RSSを用いたテレビ番組評判リアルタイム検索システム

2008MI072 石木　諒平　　2008MI217 関　智成
指導教員　河野　浩之

1　はじめに

現在 Myspace や facebook などの SNS(Social Networking Service) の普及により，容易にリアルタイムな情報を得ることが可能となった．通常のウェブ検索では，検索結果に反映されるまで数分から数日かかるのに対してリアルタイム検索は数秒から数分とタイムラグの差は明らかである．

リアルタイム性が求められるものには，株価の推移や天気の変化，交通情報などの数秒単位で情報に変化があるものが挙げられ，それらに関する検索サイトは多数存在している．また，放送中のテレビ番組について検索できるサイトや番組情報についての RSS を配信するサイトが増えてきている．本研究では，テレビ番組の情報だけでなくリアルタイムな評判も示すシステムの構築を考えた．本システムでは，リアルタイム性を追求するため，RSS リーダーを利用して，テレビ番組の情報を常時収集できるようにした．その番組の情報と，その番組に対してコメントしてあるブログを収集し，コメントの内容から評価語のスコアリングを行う．ユーザーがテレビ番組に関するキーワードを選択し，その番組についての評価情報を示すリアルタイム検索システムを実装する．

2　ブログ解析と RSS に関する諸研究

本章では，リアルタイム検索と RSS を用いた技術についていくつか紹介していく．

2.1　リアルタイム検索の現状

リアルタイム検索は，2009 年 10 月から Google によってスタートしたサービスであり，2010 年 8 月に日本で正式に公開された．

Google リアルタイム検索は，場所による絞り込みや，関連性の高いやり取りをスレッド形式で表示できるといった機能を持ち，ユーザーからの評価は非常に高かった．Google 以外に Yahoo!や Bing もリアルタイム検索を公開し始め，現在日本語でリアルタイム検索できるサイトは，Yahoo!, Bing, NAVER の 3 つである．これら 3 つのサイトで比較を行った．お菓子というキーワードによる検索結果を表 1 に示す．

Yahoo!リアルタイム検索では，検索結果の表示方法を選択できる．また，注目キーワードの表示があり，リツイート・お気に入り登録も可能である．Bing リアルタイム検索は，注目リンクがあり，リツイートが可能であるが，日本語以外もヒットしてしまう．NAVER リアルタイム検索では，twitter・ブログ・ミニブログを表示でき，トピックワードがある．

また，blog のタイトルや本文の概要等が記述された RSS(RDF Site Summary) フィードを収集して，インデックスを作成し，検索サービスを行う RSS 検索サイトがある．国内 blog を対象とした RSS 検索サイトとしては，未来検索 livedoor や Bulkfeeds が有名である．RSS 検索サイトの検索対象はさまざまな組織をわたっての blog であり，検索結果を RSS でも提供するという特徴がある．Bulkfeeds では，ping サーバの更新情報，blog ホスティングサービスの更新情報，人手による登録といった 3 種類の方法で RSS の収集を行っている．また，Feedback，Myblogjapan，ココログル，BLOGNAVI，blogsearch などでは，blog に的を絞って検索ができる．これらは，ping サーバから blog サイトの情報を得て，RSS を利用することで blog を収集する検索システムである．

表 1　検索エンジンの比較

検索エンジン	リアルタイム性	検索対象期間
Yahoo!	約 1〜5 秒	ツイート (過去 24 時間)
Bing	約 30〜60 秒	ツイート (過去 7 日間)
NAVER	約 1 分	ブログ・ツイート (過去 8 時間)

2.2　RSS を用いた情報収集

奥村ら [1] の研究では，クローリングした HTML 文書を解析し，その Web ページが blog であるかどうかの判定を行って blog を収集している．奥村らの手法は，特定のシステムや blog ツールを用いていない Web ページなども収集できること，HTML 文書を直接解析することで，過去のものまで収集することができるという利点がある．水口ら [2] は，ブログデータを定常的に収集し評判情報を検索する eHyouban という評判情報分析システムを開発した．eHyouban は，「良い」「悪い」などの評価表現だけでなく，どこが良いのかなどの注目ポイントも抽出できる点や，従来必要であった評価対象品の名前と注目ポイント単語の事前登録なしに評判情報検索を行えるという特徴を持つ．システムの構成は，ブログ収集部，評判情報抽出部，検索・分析部に分けられ，ブログ収集部でブログの RSS 情報を入手し，常時ブログの収集を行うことでリアルタイム性が実現されている．

3　番組検索システムの構成

本章では，本研究で構築するシステムについて説明を行う．

3.1　MagpieRSS のついて

今回の研究では RSS を取得するため，MagpieRSS という RSS リーダーを利用する．MagpieRSS は PHP で

利用可能な RSS パーサーであり，PHP でリモートの RSS を取得，解析，キャッシュできる．また RSS0.9 と RSS1.0 に対応している．RSS リーダーには様々なものがあり，ヘッドラインをティッカー表示するティッカー型や，ポータルサイトのマイページなどに登録するホームページ型，システムトレイに常駐して更新時に教えてくれる常駐型などがあるが，今回は PHP と連携して利用するため MagpieRSS を利用することにした．

3.2 システムの構築

まず，番組の情報を提供するサイトの RSS を購読し，常に情報を手に入れられるようにする．番組情報に更新があれば，その番組に関するブログの記事を Wget を用いて収集し，収集したブログを解析プログラムに通し，データベースに格納していく．データベースは MySQL を選択した．ここでは本研究の検索システムの構築手順を (1)〜(6) に示し，検索システムとユーザーの関係を図 1 に表示する．

(1) 更新があった場合，番組の情報とその番組に対するブログの記事を Wget を用いて収集する．この時，ブログの記事は拡張子が html のものだけ収集する．収集したブログのファイルは Perl プログラムを読み込みやすくするため，全て同一ディレクトリに保存する．

(2)Perl のプログラムを実行して，収集したブログを解析しデータベースに格納していく．この際形態素解析，評価表現抽出，評価語のスコアリング，全文検索を行うためのインデックス作成もこのプログラム内で行う．格納していく内容は，ブログ URL，評価語のスコア，全文検索を行うための全文インデックスである．

(3) ユーザーが PHP を用いて，検索したい商品をキーワード検索画面で選択する．

(4) 検索画面から選択されたキーワードを検索 PHP プログラムに送る．

(5) 送られてきたキーワードを基に，PHP 上から SQL 文をのせてデータベースにアクセスし，キーワードが MySQL に格納されているか検索を行う．

(6)PHP プログラムを用いて，ユーザーが選択した番組の情報と評判情報を出力する．

3.3 検索画面

本システムではキーワードを選択して検索する手法をとる．まず，検索画面でテレビ局を選択する．テレビ局を選択すると，その局で放送されているテレビ番組一覧が表示される．次に，評判を検索したいキーワードを入力することで検索結果ページに検索キーワードを含むブログのリンクと評価が表示されるようシステムを構築する．

検索結果の画面では以下の項目を表示する．

・番組名
・放送日
・放送時間
・放送局
・番組に対する評判
・ブログのタイトル
・ブログへのリンク

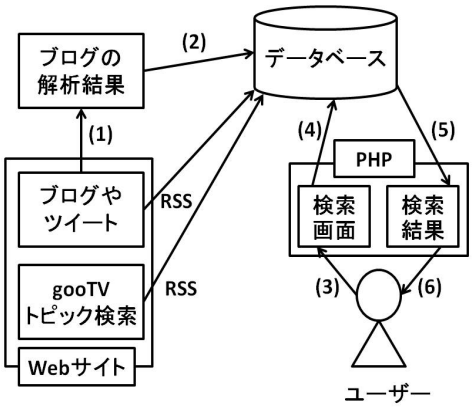

図 1　番組情報取得システムの構築

4 番組評判検索システムの実装

4.1 番組情報取得システムの各プログラム

本研究でシステムを構築する際に作成したプログラムを以下に示す．

● rss.php
番組情報に関する RSS を購読し，取得した RSS から必要なデータをデータベースへ格納する．

● blog_rss.php
テレビに関するブログから RSS を購読し，取得した RSS から必要なデータをデータベースへ格納する．

● hyoka.pl
ブログを読み込み形態素解析，評価語の抽出，評価語のスコアリングを行って結果をデータベースに格納する．

● input.php
ユーザーが検索したいテレビ局を選択する画面．

● list.php
input.php でユーザーが選択したテレビ局名をキーワードとしてデータベースにアクセス，ヒットした番組名を検索結果として一覧で表示する．

● kekka.php
list.php でユーザーが入力したキーワードでデータベースにアクセス，そのキーワードを含むブログとブログへのリンクを表示し，番組に対する評価を検索結果ページに一覧で表示する．

4.2 番組評判ブログの収集

番組は「gooTV トピック検索」のサイトから収集したものが対象であり，ソースコード取得のため，クローラの Wget を使用する．RSS リーダーで更新が確認された場合，「gooTV トピック検索」のサイトから番組の情

報を取得する．番組に関するブログは「日本ブログ村」で配信される RSS を取得し，ブログを Wget を用いて定期的に収集する．

4.3 RSS の読み込み

RSS リーダーは PHP に埋め込み可能な MagpieRSS を使用し，RSS リーダープログラムの枠内にプログラムの一部を示す．

まず，日本語が文字化けを起こさないため，スクリプト内で define('MAGPIE OUTPUT ENCODING', 'UTF-8'); を表記し，文字エンコーディングを UTF-8 に変更する．

$url で RSS を取得したい Web サイトの URL を指定する．次に fetch rss() で指定した URL から RSS を取得し，配列に入っている title, link のデータを入手する．$sql="INSERT into RSSreder.rss(title,url) VALUES ('$title', '$link2')"で title と link2 を MySQL に格納している．

RSS リーダープログラム

```
require_once('rss_fetch.inc');
define('MAGPIE_OUTPUT_ENCODING','UTF-8');
$url = 'http://tvtopic.goo.ne.jp/rss.xml';
$rss = fetch_rss($url);
$title = $rss->channel['title'];
echo"<dl>\n"
foreach ($rss->items as $item ){
  $title = $item['title'];
  $title=$my_convert_encording
      ($title,"UTF-8","auto");
  $link = $item['link'];
  echo"<dt><a href=\"$link\">
  $title</a></dt>\n";
  $sql="INSERT into RSSreader.rss
  (title,url) VALUE ('$title','$link2')"
}
```

4.4 評価語の抽出

評価表現抽出を行うプログラムは評価語の抽出プログラムの枠内のようになる．while(<HYO>) の中で，形態素解析の結果を空白で分割し，@blog の配列に格納する．形態素解析の結果の品詞の分類が「形容動詞」「形容詞-自立」「形容詞-非自立」である場合，評価語として抽出される．また，評価語が発見されなかった場合は「評価語はありません」と表示される．

次に評価語のスコアリングを行うプログラムを評価語のスコアリングプログラム枠内に示す．最初に$ward =$_; で評価語が格納されているテキストファイルから評価語を読み込む．次に，open(JISYO， $in) で鍛冶ら [3] のスコア辞書を参考にして自ら作成したスコア辞書を用いる．while(<JISYO>) で改行を削除, 空白で分割し，@data に内容を格納する．評価語の内容とスコア辞書の内容が一致した場合に$score += $data[0]; よりスコアの合計を計算している．また，もし評価語がなかった場合は「評価語ではありません」と出力される．

評価語の抽出プログラム

```
while(<HYO>){
    chomp ($_);
    @blog = split(/\s+/, $_);
for($i=0; $i<@blog; $i++){
    $blog[$i] = ~ s/( | )+//g;
    }
if(($blog[3] =~ /形容動詞/)||
   ($blog[3] =~ /形容詞-自立/)||
   ($blog[3] =~ /形容詞-非自立/)){
    $count++;
    print OUT "$blog[2]\n";
    }
elsif(($blog[0] eq "EOS") and
      ($count == 0)){
       print OUT "評価語はありません\n";
}
```

評価語のスコアリングプログラム

```
while(<HYO>){
    chomp ($_);
    $ward = $_;
    $in = "sukoajisyo. txt";
    open(JISYO， $in) or die
("can't open file $in\n");
    while(<JISYO>){
chomp ($_);
@data = split(/\s+/， $_);
for($k=0; $k<@data; $k++){
    $data[$k] =~ s/( | )+//g;}
if($ward eq $data[1]){
    $score += $data[0];
    print OUT "$ward $data[0] \n";
        }
elsif($data[1] eq "EOS"){
    print OUT "評価語ではありません\n";}
        }
    close(JISYO);
}
```

5 番組評判システムの考察評価

5.1 システムの実行例と評価

構築したシステムの実行例を以下に示す．まず，テレビ局選択ページで検索したいテレビ局を選択する．実行例では TBS を選択した．その結果を図 2 に示す．選択されたテレビ局で放送中の番組名一覧が表示されている．次に番組一覧ページの下にあるブログ検索でキーワードをドラマとして検索する．その結果，図 3 のようにドラマというワードを含むブログ文のタイトルとリンクが表示される．表示されたブログはスコアリング結果が良い順に一覧で並べられる．

「gooTV トピック検索」では，番組の情報のみ表示されていたが，本研究ではブログの検索機能が追加され

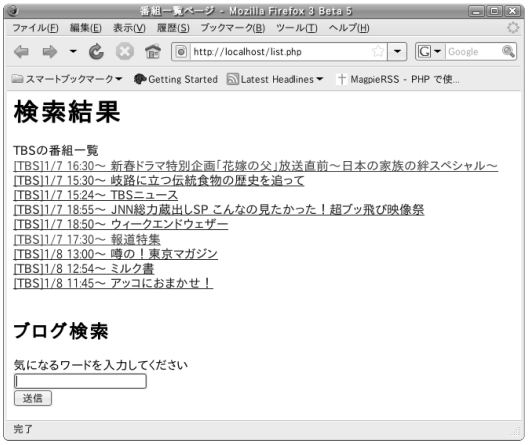

図 2　番組一覧ページの実行例

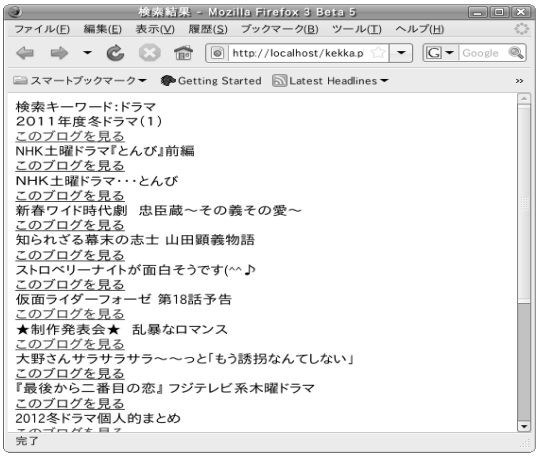

図 3　ブログ検索結果ページの実行例

た．また，原田ら [4] の研究ではブログの収集を手動で行っていたが，本研究では RSS を利用したことにより，効率よくブログを収集できる．

5.2　評判検索における RSS の利用価値

本研究では RSS を利用し，番組の情報を取得した．RSS を利用することによって，リアルタイム性を重視したシステムの構築が可能となり，新たなサービスが行えるようになると考える．「gooTV トピック検索」は，番組の情報を検索する機能を持つ．これに対して，本研究で構築したシステムは番組の情報とともにその番組に対する評判を示し，番組に関するブログへのリンクを検索結果に追加した．テレビ番組の情報や評判をリアルタイムに取得できることによって，現在どのような番組が視聴者からの評価が高いかを知ることができ，人気番組の情報をすぐに手に入れることができる．

5.3　本研究の応用と課題

本研究では検索対象のデータをテレビ番組としてシステムを構築した．テレビ番組のデータは評判検索システムで用いることのできる対象データの一例である．本研究では番組情報とブログを RSS で取得した．よって，対象の RSS を変更することで様々なデータに関する評判検索システムを構築できる．例えば，音楽やファッションに関する評判検索についてのシステムに応用できると考える．

今後の課題としては，RSS で更新情報を取得後，自動で blog を収集し解析を行えるようシステムを構築する必要があり，毎週放送される番組で同タイトルでも毎回，評価内容が異なるため，区別ができるようにシステムを検討しなければならない．また，本研究で構築したデータ辞書に格納されている評価語が番組の内容を指す文章に含まれていた場合，正確な評価が行われなかったためその点の改善も必要である．

6　まとめ

本研究では，Web 上に存在する未整理のブログからの番組情報の収集が難しいという問題点を以下のシステムによって解決を試みた．形態素解析ツールの ChaSen を使用し，ブログに書かれている評価を抽出する．評価表現の抽出は，抽出した単語を辞書と照合することにより，スコアリングを行った．ユーザーがキーワードを選択し，それについての記述があるブログを検索結果ページに表示した．また，番組情報と番組に関するブログを取得する部分では，MagpieRSS を用いて自動でデータベースに格納できるようにシステムを構築した．ブログ解析システムにおける RSS の利用は，不定期に更新されるブログから自分が必要とするブログのみを効率よく収集できる利点があるといえる．

今後の課題として，状況に応じて評価表現の更新を行えるようにし，実用化のためにデザインについて検討する必要がある．

参考文献

[1] 奥村学，南野朋之，藤木稔明，鈴木泰裕，“ blog ページの自動収集と監視に基づくテキストマイニング，”人工知能学会研究会資料，pp.1-8，2004.

[2] 水口弘紀，土田正明，“ Weblog を対象にしたリアルタイム評判情報分析システム eHyouban，”DEWS2008，I2-27.

[3] 鍛治伸裕，喜連川優，“ 自動構築した評価文コーパスからの評価表現辞書の構築，”日本データベース学会 Letters Vol.6, No.1, pp.1-4, 2007.

[4] 原田健太，堀山洋輔，“ ジオコーディング技術による釣りブログの可視化，”南山大学数理情報学部 情報通信学科 2010 年度卒業論文要旨集，pp.170-173，2011.