

ソーシャルメディアを含む 多メディアビッグデータの統合的解析による情報抽出

上田高德^{†1} 浅井洋樹^{†1} 藤木紫乃^{†1} 山本祐輔^{†1} 武井宏将^{†1,†2} 秋岡明香^{†1}
山名早人^{†1,†3}

本稿では我々が取り組んでいる多メディアビッグデータの統合的解析による情報抽出の試みについて述べる。ソーシャルメディアの普及によって、様々な情報がリアルタイムにインターネット上にアップロードされるようになった。我々は、単一のソーシャルメディアだけでなく、複数の情報源を組み合わせた、「多メディアデータ」を解析することで、より有益な情報を抽出できると考えている。本稿では我々が取り組んでいる多メディア解析について述べる。また、大規模リアルタイムデータの解析をサポートするために開発している、並列分散処理フレームワーク QueueLinker についても述べる。

1. はじめに

Twitter をはじめとするマイクロブログの普及により、多くの人々が情報をリアルタイムにインターネット上へアップロードするようになった。Twitter が持つ特性の解析も進み 1)2), いまや Twitter は貴重な情報メディアと考えられている。Twitter の価値はその内容の豊富さだけでなく、リアルタイム性の高い投稿が多い点にあり、Twitter データを解析することで、リアルタイム性の高い情報を抽出することが可能になる。たとえば、Twitter の投稿から地震の発生をリアルタイムに検知することができる 5)。

我々もこれまで、ソーシャルメディアを用いたデータ解析に取り組んできた 4)8)10)12)。我々は、より有意義な情報を抽出するために、単一のソーシャルメディアだけでなく、Twitter と TV 番組情報のように複数の情報メディアを利用した解析を試みている。たとえば、ツイートとテレビ字幕情報を結びつけることで、どの Twitter ユーザがどのテレビ番組を見ているか判定することができる 12)。これは、番組を鑑賞している Twitter ユーザは、番組の感想を Twitter に投稿する際に登場人物の名前や役者、台詞をツイート内に含める可能性があるからである。この解析をツイートデータのみで試みた場合、テレビ番組内容に関する情報が利用できないため、より判定が難しくなる。本稿では、複数の情報メディアを組み合わせる解析を「多メディア解析」と呼ぶ。

本報告では我々の多メディア解析に関する取り組みについて説明する。我々は Web ページデータや Twitter データ、テレビ番組情報を、統合的に解析することで情報を抽出することを目指しており、これまでにいくつかの事例が得られている。また我々は、多メディア解析をリアルタイムに行うための処理基盤として、リアルタイム並列分散処

理フレームワーク QueueLinker 6) を開発している。そして、並列分散 Web クローラを QueueLinker 上で動作させることに成功している 7)。収集した Twitter データや Web ページデータを QueueLinker 上でリアルタイム処理することにより、大規模リアルタイムデータからの情報抽出を目指している。

本稿では以下の構成をとる。2 節で我々が解析に利用しているデータの概要について述べる。3 節において多メディア解析について実例を述べる。4 節でリアルタイム多メディア解析のために開発されている並列分散処理フレームワーク QueueLinker について述べ、5 節でまとめる。

2. 収集中のデータとその特性

本節では、多メディア解析のために収集しているデータと、そのデータの特性について述べる。我々は Twitter とテレビ関係データの蓄積を進めている。本節では特に Twitter のデータ収集と、得られた Twitter データの特性について述べる。Twitter のようなソーシャルメディアのデータ取得においては、API からのデータ取得量が制限されている場合が普通であり、多くのデータを用いて解析するためには、継続的にデータを収集する必要がある。

2.1 Twitter データ収集

Twitter からデータを収集する主な方法として、Sample API と Filter API がある。Sample API は全ツイート中からサンプリングされたツイートを取得でき、Filter API は登録した単語を含むツイートを取得することができる。我々は図 1 のように計算機を用いて、Sample API と Filter API の双方を利用して Twitter データを収集している。

2011 年 12 月 4 日～2012 年 11 月 8 日の間に Sample API を用いて収集できたツイートデータは、合計で約 9 億 4200 万ツイートであった。Sample API で取得できるツイートに加えて、話題の変化に応じたツイートを収集するために、時期に応じた様々な単語を Filter API に登録して収集してきた。これまでに収集に用いた単語例を表 1 に示した。

^{†1} 早稲田大学
Waseda University
^{†2} 日本ユニシス株式会社
Nihon Unisys, Ltd.
^{†3} 国立情報学研究所
National Institute of Informatics

URL を含むツイートを収集するために、「http」や「https」を含むツイートも収集している。このほか、ツイートデータの提供を行っている Gnip 社より、2012 年に行われたロンドンオリンピック期間を含む、2012 年 7 月 27 日～8 月 14 日の約 8,850 万ツイートの提供を受けた。

以上、これまでに蓄積したツイートの合計データサイズは約 8TB になる。

2.2 得られた Twitter データの特性

次に、収集した Twitter データの特性について検討し、多メディア解析の可能性について検討する。

2.2.1 テレビ実況ツイート

テレビ番組を見ながら番組の感想を掲示板などに投稿する行動はテレビ実況と呼ばれ、以前よりインターネット掲示板を中心に行われてきた。掲示板に投稿される実況を利用して番組の盛り上がり場面を抽出する [11] など、テレビ実況には大きな利用価値がある。

近年は Twitter の登場により、さらに気軽にテレビ実況を行える環境になったといえる。2011 年 12 月 5 日～2012 年 11 月 12 日の期間で、表 1 のテレビ実況の欄で示したハッシュタグを用いて Filter API で収集できたツイート数は約 3,520 万であった。2011 年 12 月 9 日～2012 年 11 月 12 日の

期間で、表 1 のアニメ実況の欄で示したハッシュタグで収集できたツイート数は約 1,830 万ツイートであった。多くのテレビ実況ツイートが Twitter に投稿されていることが分かる。

しかし、Twitter に投稿できるのは 140 字までであるため、Twitter の投稿には、どのテレビ番組についてのツイートなのか明記しないことが多い。また、ハッシュタグを付けずに番組について述べているツイートもあるため、ハッシュタグのみでは検出できない実況ツイートもある。ツイートが実況ツイートであるかどうか、また実況ツイートであるならどの番組に関するものか判定することは、チャレンジングな課題といえる。

この課題に対するアプローチとして、テレビ番組に登場する人物名や台詞の情報を用いて、ハッシュタグが付加されていない実況ツイートの抽出を試みる事が考えられる。番組を鑑賞している Twitter ユーザは、番組の感想を Twitter に投稿する際に登場人物や役者の名前、台詞をツイート内に含める可能性があるためである。ここに、複数のメディアデータを組み合わせて解析する意味がある。実例は 3.1 で述べる。

2.2.2 URL を含んだツイート

ツイートの投稿には外部情報を参照するために、URL が含まれることがある。URL を含むツイートを収集するために、2012 年 10 月 10 日～11 月 8 日において、「http」や「https」を含むツイートを Filter API で収集した。この期間において収集できたのは約 8,450 万ツイートであった。

図 2 は URL を含むツイートに対して、各ツイートの URL 部分を除いた文字数の頻度分布を示したものである。図 2 から、URL のみでなく、何らかの付帯情報と共にツイートが行われていることを確認できる。この中には URL が指す Web ページの概要や内容に対する感想を含んでいるものがあると考えられる。つまり、Twitter ユーザが Web ページの属性付けを行っているとも考えられ、Twitter が Web ページ

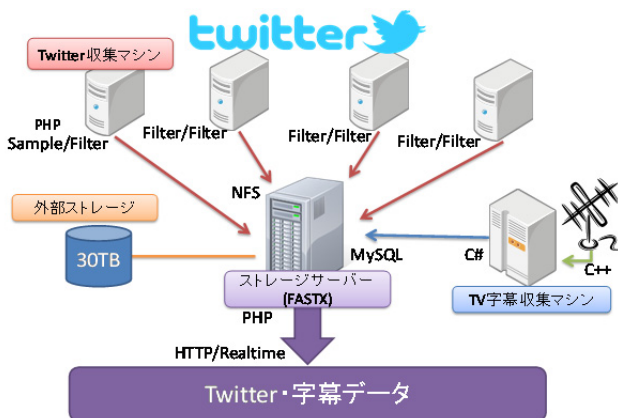


図 1 ツイートデータ収集の計算機接続図

表 1 Filter API での収集内容

収集目的	登録フィルタ単語
リツイート	RT
URL 付きツイート	http, https
テレビ実況	#nhk, #etv, #ntv, #tvasahi, #tbs, #tvtoyko, #fujitv, #tvk, #tokyomx
アニメ実況	#laputa, #guiltycrown, #ungo, #fatezero, #kyoukaisen, #lastexile, #P4_A, #tamayura, #ben_to, #Wagnaria, #C3_anime, #kimiboku, #mirainikki, #future_diary, #ikamusume, #geso, #mashiro_tv, #g_age, #haganai, #hunterhunter, #chihaya_anime, #hidamari, #majikoi, #makenki, #shana, #SEKAKOI
Foursquare	I'm at
コミックマーケット 81	#c81
ガキの使いやあらへんで 2011 大晦日 SP	gaki, gakisp, ntv
紅白歌合戦 2011	nhk, kouhaku, nhk_kouhaku61, 紅白
金環日食	金環日食
オリンピック関係	#london2012, #olympic, #オリンピック, #アーチェリー, #ウエイトリフティング, #カヌー, #近代五種, #競泳, #サッカー, #射撃, #シンクロ, #新体操, #柔道, #自転車, #水球, #セーリング, #体操, #卓球, #テニス, #飛込, #テコンドー, #トライアスロン, #トランポリン, #ハンドボール, #馬術, #バスケット, #バレー, #バドミントン, #ビーチバレー, #フェンシング, #ホッケー, #ボート, #ボクシング, #陸上, #レスリング

に対するソーシャルタギングシステムとして機能している
 と考えることもできる。URL 付きのツイートを利用するこ
 とで、Web ページの内容を的確に把握できる可能性もある。

他に、類似した内容の Web ページについて言及している
 ユーザは興味対象が近いと考えられるから、ユーザ間の興
 味類似度といった解析が可能になるとも考えられる。これ
 は、Twitter と Web データを組み合わせた多メディア解析の
 可能性といえる。

なお、図 2 において 116 字、120 字にピークができてい
 るのは、ホームページにあるツイートボタンなどでツイ
 ートする際に、入力したコメントが Twitter の文字数制限を超
 過されたために削除され、該当の文字数に切りつめられた
 ツイートが多くあるためと考えられる。

2.3 その他に収集を試みているデータ

Twitter データのみでなく、Web ページ構造と組み合わせ
 て解析するために、大規模 Web データの収集・解析の検討
 を進めている。我々は大規模データをリアルタイムに解析
 するために、並列分散フレームワーク QueueLinker を開発
 しており、QueueLinker 上で動作する並列分散 Web クロー
 ラを開発している 7)。詳細については 4 節で説明する。

3. 多メディア解析の事例

本節では、我々が行っている多メディア解析の個々の実
 例について述べる。

3.1 テレビ番組の視聴者判定

2.2.1 で述べたように、Twitter 上には多くのテレビ実況ツ
 イートがある。Twitter データから、各ユーザがどのテレビ
 番組を見ているか判定することができれば、視聴率調査や
 番組の人気度の測定など、様々な社会的調査への貢献が期
 待でき、これまでも実況ツイートの検出が試みられている
 9)。ビデオリサーチ社もツイートから視聴率を抽出するこ
 とを試み初めており 14)、その重要性が注目されている。

我々は、テレビ字幕情報とツイート情報を用いることで、
 視聴者判定が行えることをすでに示している 12)。番組を
 鑑賞している Twitter ユーザは、番組の感想を Twitter に投
 稿する際に登場人物や役者の名前、台詞をツイート内に含
 めると期待される。そこで、TV 字幕の内容とツイート内容
 のマッチングを取ることで、精度よく視聴者判定を行う
 ことができる。

3.1.1 判定システムとアルゴリズム概要

図 3 に判定システムの概要図を示した。字幕テキストか
 らテレビ番組に特徴的な名詞を抽出し、ツイート中に抽出
 した特徴語が含まれた場合にテレビ番組を視聴していると
 判定する。ここでは判定アルゴリズムの概要を説明する。

字幕テキストを取得できたテレビ番組の総数を P とす
 る。テレビ字幕に含まれるある名詞 $term_i$ を考えたとき、
 $term_i$ が字幕中に出現した番組の数を $p_i \geq 1$ とする。い
 ま、 $term_i$ の idf 値を以下のように定義する。

$$idf_i = \log \frac{P}{p_i}$$

ここで、 idf 値が大きい名詞はテレビ番組における特徴
 的な名詞といえ、その名詞を含むツイートを投稿したユー
 ザはテレビ番組を視聴している可能性がある。逆に、 idf 値
 の小さな名詞は一般名詞と考えられる。そこで、閾値より

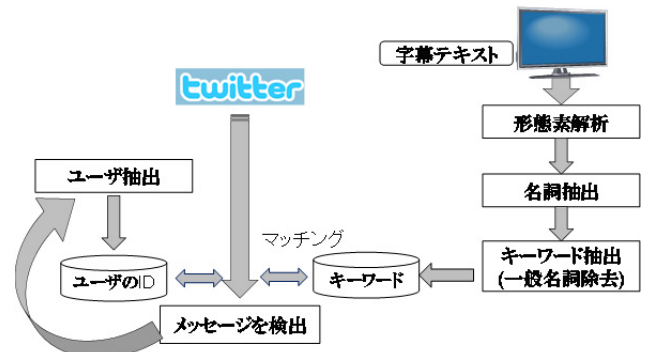


図 3 視聴者判定システム概要

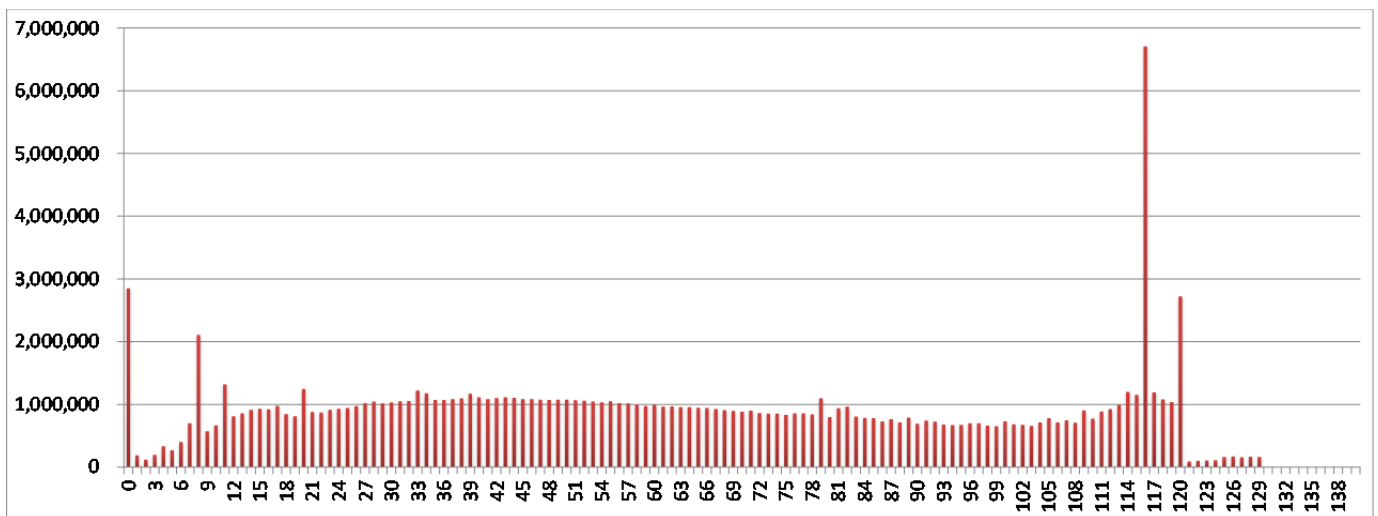


図 2 URL を除いた部分の文字数の統計

も小さい名詞を字幕テキストより削除して、残った名詞を特徴語とする。そして、取得したツイートと特徴語を比較し、ツイートに特徴語が含まれていたらテレビ番組を視聴していると判定する。

3.1.2 抽出結果

本システムで抽出できたツイートの一例には以下のようなものがある。

- 日本のメーカーはわかってないな。ワンセグや高画質カメラより、いかに使える。楽しいアプリがあるかだよ。

このツイート内容そのものには、テレビを視聴していると推定できる直接的な情報を含んでいない。しかし、この時間帯にスマートフォンに関するドキュメンタリーが放送されており、本システムで抽出した特徴語に基づいて検出することが可能になっている。これは多メディアデータ解析が効果を生む例といえる。

3.2 TV 実況解析 bot (@jikkyo_analyzer)

本項では、実況ツイートを解析して、テレビ番組を端的に表現する単語を抽出する試みについて説明する。解析で得られた単語は TV 実況解析 bot (@jikkyo_analyzer) において Twitter 上でサービスを提供している。

本システムの全体の構成を図 4 に示した。本システムでは番組が変わるごとにツイートデータの統計を取り、tf-idf ライクな手法により特徴語を抽出している。特徴語は番組ごとに異なるため、テレビ番組情報と組み合わせて利用することで、解析対象のツイート範囲を番組の変化に合わせて行うことが可能になる。

3.2.1 抽出手法

ここで、TV 実況解析 bot の抽出方法について概要を述べる。Twitter Filter API を用いて、表 1 に示されたテレビ実況に関するハッシュタグが付いたツイートを収集する。収集できた全ツイート集合を A とする。また、あるテレビ局の番組 v の実況ツイート集合を $t_v \subset A$ とする。ある名詞・形容詞 $term_i$ を考えたとき、 t_v に含まれるツイートのうち $term_i$ を含むツイート数を tf_i 、全実況ツイート A に含まれるツイートのうち $term_i$ を含むツイート数を af_i とする。全収集ツイート数 $N = |A|$ として、以下の式を用いてスコアを算出する。

$$w_i = tf_i \log \frac{N}{af_i}$$

このスコア w_i の高い単語を番組 v に関する特徴的な単語と考え、 w_i の上位から順に Twitter の文字数制限を超えない範囲で、Twitter に自動投稿する。

3.2.2 抽出結果

例として、オリンピック期間中に抽出できた単語情報を表 2 に示した。開会式や競技に関する特徴的な単語が取得できていることが分かる。番組ごとのデータを用いることで、オリンピックの進行に合わせて抽出が行えた。表 2 中

の「勢い〇〇Pt」とは、その番組に対する実況ツイートの数を表している。本 bot のもう一つの機能として、1 時間毎に各テレビ局の実況ツイート数を集計してツイートしている。これにより、最近 1 時間で実況が盛り上がったテレビ局がわかる。このほか、本システムの単語抽出結果は、13) から日付順に参照できる。

3.3 新たに取り組んでいる多メディア解析

本項では、我々が新たに取り組んでいる多メディア解析について述べる。現在、実況ツイートに関してさらなる解析を行うために、テレビ局のハッシュタグと URL の双方が含まれているツイートに注目している。放送中の URL 付きツイートは、テレビ放送に関連した URL を提示している可能性がある。従って、ツイートに含まれる URL が指す Web ページを解析することで、テレビ番組の補足情報を得られる可能性がある。さらに、Web ページに含まれる画像を抽出して分類することで TV 番組を補足する画像情報を得ることを試みている。

また、類似した内容の Web ページについて言及している Twitter ユーザは、類似したページに対して興味を持っていることから、興味対象が近い可能性がある。Web ページ内容と組み合わせることで、Twitter 上でのフォロー

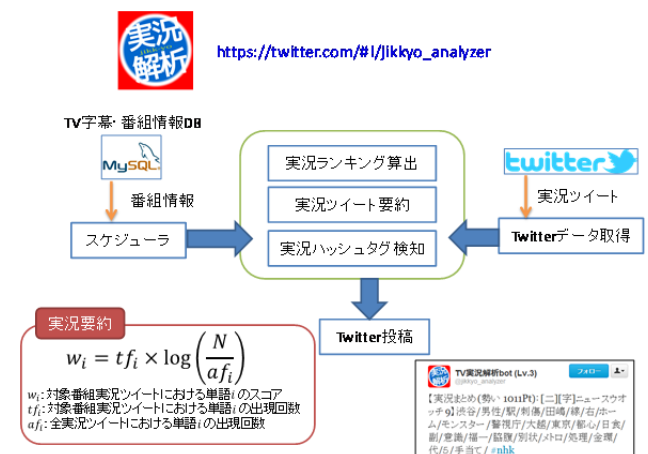


図 4 実況解析 bot の構成図

表 2 実況解析 bot による抽出例

日時	抽出内容
2012/7/28 09:07:09	【実況まとめ(勢い 4370Pt) : 週刊 ニュース深読み▽いよいよ開幕！ロンドン五輪】聖火/チャリで来た/傘/ハト/陛下/女王/ヘリ/ベッカム/メリー・ポピンズ/船/オリンピック/開会式/ポール/12/20/ら/ロンドン五輪/NHK/ #nhk #olympic
2012/7/28 19:31:39	【実況まとめ(勢い 301Pt) : NHK ニュース 7&オリンピック【二【字】】NHK/スウェーデン/自転車男/RR/日本/なでしこ/オリンピック/子/個人/tvasahi/柔道/内乱/美名/小郷/フルーム/自/競泳/#nhk
2012/7/28 21:57:32	【実況まとめ(勢い 2315Pt) : ロンドンオリンピック 2012 柔道予選・競泳予選ほ】オリンピック/柔道/tvasahi/Olympic/バレー/アルジェリア/日本/平岡/女子/セット/キロ級/福見/本/なでしこ/NHK/サッカー/勝ち/ #tvasahi #オリンピック

関係では直接つながっていない、類似ユーザクラスを抽出できる可能性がある。発表ではこれらの解析の試みについても説明する予定である。

4. 並列分散処理フレームワーク QueueLinker

Twitterをはじめとするマイクロブログはリアルタイムな情報源であるから、リアルタイムに処理を行わなければ情報の価値が損なわれる可能性がある。多メディア解析においてリアルタイム処理を行う場合、その大きなデータ量のために並列分散処理が必要になる場合が多い。

本節では、リアルタイムな多メディア解析を実現するために、我々が開発している並列分散処理フレームワーク QueueLinker について概要を説明する。QueueLinker は Producer-Consumer モデルをプログラミングモデルに採用している。モジュールを Producer-Consumer モデルを用いて実装し、モジュール間の接続関係を QueueLinker に与えると、QueueLinker はモジュールのインスタンスを計算機に割り当て、処理を並列分散実行する。モジュール間のデータ転送は QueueLinker が実行するため、プログラマは通信処理を記述することなく並列分散処理を実行できる。現在は初出の原稿 6) から改良が進み、様々なアプリケーションを実行できるようになってきている。例として、我々が開発している Web クローラ 7) がある。図 5 に Web クローラのモジュール接続図を示した。

4.1 プログラミングインターフェース

本項では QueueLinker のプログラミングインターフェースについて概要を述べる。Algorithm 1 にモジュールの基本実装方法を示した。Algorithm 1 は入力キューを N_{in} 個、出力キューを 1 個持っているモジュールの例である。どのキューに入力されたデータかは id で識別できる。入力キューに応じた処理を行い、結果のオブジェクトを返すことで、QueueLinker が次のモジュールに転送する。モジュール間

で内部状態を共有しないようにし、キューのみでデータをやり取りとりして処理を進めるように実装すれば、各モジュールは任意の計算機において実行できることになる。また、キューでやり取りするデータのハッシュ値で計算空間を分割できるのであれば、各モジュールは任意のスレッド数で並列実行できる。

このような形式で実装されたモジュールの接続関係を QueueLinker に与えると、QueueLinker が各モジュールを計算機に割り当て自動的に並列分散実行する。分散処理においてはデータのハッシュ値に基づいて計算機間で処理範囲の分割を行うことができる。

4.2 並列分散スケジューラの構成の試み

与えられる図 5 のようなグラフはモジュールの論理的な接続関係を表わしているから、各モジュールにどれほどの計算リソースを割り当てて処理を実行するか調整が研究課題になる。現状ではモジュールごとに使用する計算機やスレッド数について手動での指定が必要であるが、自動スケジューリング機構の実装を進めており、今後報告予定である。

4.3 多メディア解析への応用

本項では QueueLinker の多メディア解析への応用例を説

Algorithm 1 Module Example

```
1: procedure MODULE(data, id)
2:   if id = 0 then
3:     data に対して処理 0 を実行
4:   else if id = 1 then
5:     data に対して処理 1 を実行
6:   (... 中略...)
7:   else if id =  $N_{in} - 1$  then
8:     data に対して処理  $N_{in} - 1$  を実行
9:   end if
10:  return data
11: end procedure
```

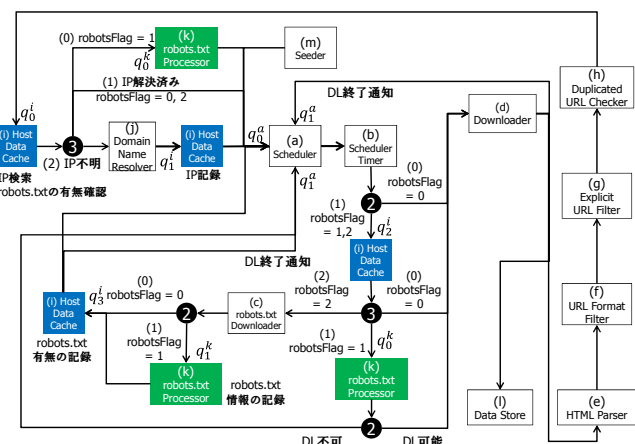


図 5 Web クローラのモジュール接続図

a 多入力、多出力のモジュールもサポートしているが本稿では省略する。

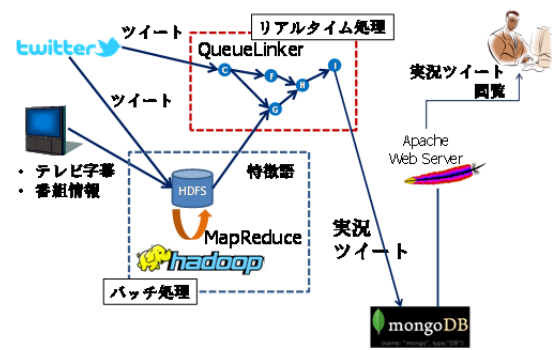


図 6 QueueLinker によるリアルタイム視聴者判定



図 7 リアルタイム画像分類システムの構成図

明する。3.1 で説明した視聴者判定をリアルタイムに行うために、図 6 のような構成で処理を行った。Hadoop を用いてバッチ処理を行うことで特徴語を抽出し、抽出した特徴語を QueueLinker に与えて視聴者判定を行う。抽出したツイートを mongoDB に格納し、ユーザに結果を提示する。

また、我々が開発している Web クローラと組み合わせて、Web データをリアルタイムにクローリングして、画像を分類する試みを行っている。図 7 にモジュールの接続図を示した。図 7 中にある Web クローラは図 5 に示したものと同じである。Web クローラの詳細説明は 7) にあるが、本システムでは 1 階層のみクローリングして画像のみを収集するよう修正を加えている。

図 7 の構成ではまず、Twitter Filter API を用いて、表 1 にあるテレビ実況のハッシュタグと URL を含むツイートを抽出している。ツイートに含まれる URL は短縮 URL であるので、URL を展開したうえで Web クローラに渡す。クローラは URL が示す Web ページに含まれている画像のみをクローリングする。クローリングした画像を我々が開発した画像分類手法 3) を用いて分類する。最後に、画像分類結果を蓄積する。

5. おわりに

本稿では我々の多メディア解析の取り組みについて述べた。我々が解析に利用している Twitter データの概要と、テレビ実況ツイートの解析を例に多メディア解析について述べた。そして、並列分散実行基盤である QueueLinker について概要を述べ、多メディア解析への応用例を述べた。本稿で述べたように、多メディアを統合的に解析することで、単一メディアのみでは抽出できない、有用な情報を抽出できる可能性がある。本稿が多メディア解析を検討するきっかけになれば幸いである。

謝辞 本研究は、文部科学省「Web 社会分析基盤ソフトウェアの研究開発」によるものである。

参考文献

- 1) Haewoon Kwak, Changhyun Lee, Hosung Park, and Sue Moon, "What is Twitter, A Social Network or A News Media?," In *Proc. of the 19th Int'l Conf. on World Wide Web (WWW)*, pp.591-600, Apr. 2010.
- 2) Jianshu Weng, Ee-Peng Lim, Jing Jiang, and Qi He, "TwitterRank: Finding Topic-sensitive Influential Twitterers," In *Proc. of the 3rd ACM Int'l Conf. on Web Search and Data Mining (WSDM)*, pp.261-270, Feb. 2010.
- 3) Shan-Bin Chan, Duy-Dinh Le, Shin'ichi Satoh, Hayato Yamana, "Ranking Image Annotation Using Vector Space Model," 第 4 回データ工学と情報マネジメントに関するフォーラム (DEIM 2012), 2012 年 3 月.
- 4) Shino Fujiki, Hiroya Yano, Takashi Fukuda, Hayato Yamana, "Retweet Reputation: A Bias-Free Evaluation Method for Tweeted Contents," In *Proc. of the 1st Int'l Workshop on Social Innovation and Social Media (SISoM 2011 in conjunction with ICWSM 2011)*, Jul.

2011.

- 5) Takeshi Sakaki, Makoto Okazaki and Yutaka Matsuo, "Earthquake Shakes Twitter Users: Real-time Event Detection by Social Sensors," In *Proc. of the 19th Int'l Conf. on World Wide Web (WWW)*, pp.851-860, Apr. 2010.
- 6) 上田高德, 片瀬弘晶, 森本浩介, 打田研二, 油井誠, 山名早人, 「QueueLinker: パイプライン型アプリケーションのための分散処理フレームワーク」, 第 2 回データ工学と情報マネジメントに関するフォーラム (DEIM 2010), 2010 年 2 月.
- 7) 上田高德, 佐藤亘, 鈴木大地, 打田研二, 森本浩介, 秋岡明香, 山名早人, 「Producer-Consumer 型モジュールで構成された並列分散 Web クローラの開発」, 第 5 回 Web とデータベースに関するフォーラム (WebDB Forum 2012), 2012 年 11 月.
- 8) 奥谷貴志, 藤木紫乃, 山名早人, 「プロフィール情報とフォロー関係を組み合わせた Twitter ユーザコミュニティの抽出手法」, 第 3 回ソーシャルコンピューティングシンポジウム (Soc2012) 2012 年 6 月.
- 9) 小林 尊志, 野田 雅文, 出口 大輔, 高橋 友和, 井手 一郎, 村瀬 洋, 「Twitter における実況書き込み検出手法の検討」, 電子情報通信学会技術研究報告(MVE), MVE2010-45, pp.129-130, 2010 年 6 月.
- 10) 高橋雄太, 片岡義雅, 浅井洋樹, 山本祐輔, 秋岡明香, 山名早人, 「繰り返し表現を含んだ感情的なツイートの抽出」, 第 4 回データ工学と情報マネジメントに関するフォーラム (DEIM 2012), 2012 年 3 月.
- 11) 宮森 恒, 中村聡史, 田中克己, 「番組実況チャットに基づく視聴者視点を利用した放送番組のビュー生成」, DBSJ Letters, Vol.4, No.1, pp.93-96, 2005 年 6 月.
- 12) 山本祐輔, 及川孝徳, 山名早人, 「字幕テキストの利用によるマイクロブログからのテレビ番組に言及したメッセージ検出手法」, 第 3 回データ工学と情報マネジメントに関するフォーラム (DEIM 2011), 2011 年 3 月.
- 13) TV 実況解析 bot (Lv.4)(@jikkyo_analyzer)/2012 年 07 月 27 日 - Twilog, http://twilog.org/jikkyo_analyzer/date-120727.
- 14) ビデオリサーチ Twitter 上の指標整備に着手, <http://www.videor.co.jp/press/2012/121023.htm>.