RSS 自動生成のためのタイトル生成

南野 朋之
東京工業大学大学院総合理工学研究科
nanno@lr.pi.titech.ac.jp

奥村 学
東京工業大学精密工学研究所
oku@pi.titech.ac.jp

1 はじめに

Web 上の情報が，人が目で見て理解する情報であるのに対し，近年，計算機で直接扱うことができるメタデータを付加しようという動きが活発になってきている．その一つが RSS[1] (RDF Site Summary / Really Simple Syndication) であり，現在爆発的に広がっている Weblog では，ほぼ標準で RSS によるメタデータ配信が行われるなど，広く利用されている．また RSS は，サイトの更新情報 (見出し，要約，本文，更新日時など) を配信するのに適したフォーマットであるため，多くの新聞社の Web サイトやニュースサイトなどでも利用されており，今後も RSS を配信するサイトの数は増え続けると考えられる．

しかしながら，RSS によりメタデータを配信しているサイトは，CMS(Content Management System) を利用しているような一部のサイトに限られているのが現状である．なぜなら，CMS を使えば，ユーザは HTML 文書を生成すると同時に，自動的に RSS を生成することができるが，人手で作成している Web ページなどでは，RSS Feed も人手で作成しなければならず，これにはコストがかかるためである．

そこで本研究では，HTML 文書中に含まれる時系列情報を自動的に発見し，RSS Feed を自動的に生成する手法を提案する．本稿では，その際に必要となる，Web 上の大量の RSS Feed を利用したタイトル部分の発見，及び，本文からのタイトルの生成を中心に述べる．

2 HTML 文書からの RSS の自動生成

本節では，日付表現を伴って時系列を記述する Web ページから，RSS を生成する手法について概要を述べる．

RSS Feed は，channel 要素と item 要素の二種類の情報から構成される [1, 2]．channel 要素では，Feed のタイトル，URI，概要など，Feed 全体に関する情報, 及び，含んでいる記事の一覧などが記述される．item 要素では，channel 要素で一覧を示した各記事について，記事のタイトル，URI，概要などが記述される．

よって，Web ページ中に含まれる時系列情報から RSS を自動生成するためには，適切な item 要素 (言い換えれば，“記事の範囲”) を抽出しなければならない．本研究では，図 1 に示す二つのタイプの時系列情報を対象とする．図 1(左) に示す日付ベースの時系列では，各記事に一つずつ日付表現が含まれる構造である．一方，

図 1: 日付ベース (左) と記事ベース (右) の時系列

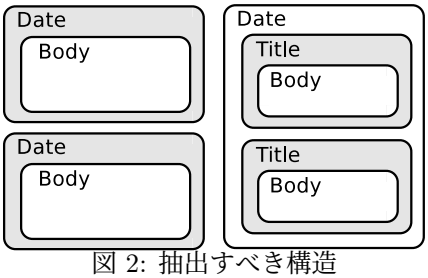

図 2: 抽出すべき構造

図 1(右) に示す記事ベースの時系列では，一つの日付表現に複数の記事が含まれる構造を示す．

以上のことから，本研究では，Web ページから図 2 に示すような構造を抽出することで，RSS を自動生成する．南野ら [3] では，HTML 文書中の日付表現を検出し，発見された日付表現をちょうど一つずつ含むようなセグメントを発見することで，図 2(左) の日付ベースの構造を自動抽出する手法を提案した．しかしながら，この手法のみでは，図 2(右) のような記事ベースの構造を扱うことができず，図 1(右) の例では,「本文 1」と「本文 2」が一つの記事として扱われてしまっていた．そこで，本研究では，記事ベースのタイプの構造を抽出可能にするために，日付表現と同様，タイトルに注目する．なぜなら，タイトル部分を発見することができれば，南野ら [3] の手法を日付表現に適用した後，再びタイトル部分に適用することで，記事ベースの構造を抽出することが可能であるからである．

また，日付ベースの時系列情報には記事のタイトルが含まれないため，RSS Feed を生成する際に，適切なタイトルを生成する必要がある．なぜなら，記事のタイトルは，item 要素の必須項目であると同時に，ユーザがページを訪れて記事の本文を読むかどうか判断するための重要な材料になるためである．

よって，次節以降では，時系列を記述する Web ページから RSS Feed を自動生成する際に必須のタスクとなる,「タイトルの発見」と「タイトルの自動生成」の二つのタスクについて述べる．

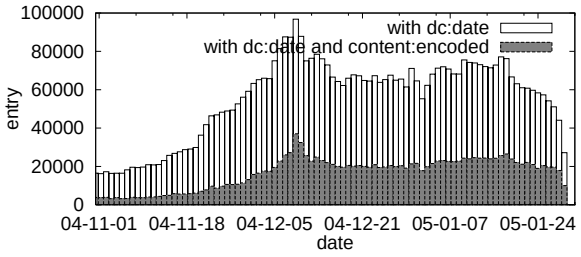

図 3: 収集した記事数の推移

2.1 RSS の収集

タイトルの発見，タイトルの生成は，大量に収集した RSS から得た統計情報を利用して行う．

収集した RSS Feed は，`ping.bloggers.jp`[1]で公開されている changes.xml の情報を元に収集した，主に Weblog の RSS Feed である．

収集したデータは，一日あたり約 6 万〜7 万記事[2]．記事の日付分布は，図 3 のようになっている．なお，図 3 は，記事の書かれた時間を示す要素 “dc:date [3] ” と，さらに，そのうち本文全体を含む要素 “content:encoded [4] ” を持つ記事数の分布を示している．

以降では，このように収集した RSS Feed から得られる統計情報を利用したタイトルの発見とタイトルの自動生成について述べる．

2.2 タイトルの発見

本節では，タイトルの発見について述べる．なお，この処理を適用する時点までに，システムは Web ページ中の時系列情報を発見し，日付ベースの構造を抽出しているとする．また，以下の二つの仮定をおく．

1. タイトルは本文の前にある
2. すべてのタイトルは単一のタグ[5](属性を含む) でマークアップされている

二つ目の仮定については，多くのケースに当てはまるが，例外も存在する．例えば，タイトルだけをマークアップしているタグが存在せず，タイトルに付いている飾り文字など，タイトル以外の要素を含めてマークアップされているケースがある．しかし，本研究ではこれらがタイトルに混入することは，RSS Feed を作成する目的では，それほど問題にならないと考え，この仮定を導入した．

以上のような仮定を行うと，タイトルを発見する問題は，タイトルをマークアップしているタグを発見するという問題に置き換えて考えることが可能になる．そこでシステムはまず，以下の条件を満たす，タイトルをマークアップしている可能性のある候補タグを選択する．

1. すべての日付単位の記事に最低一度以上出現する
2. そのタグでマークアップされている範囲に，text が存在する
3. そのタグでマークアップされている範囲に blockquote, center, div, h1..6, p, li, td, hr など改行効果のあるタグが出現しない
4. 日付表現と，そのタグの中で最初に現れるタグ間に出現する text が 200byte 以下．(日付と最初のタイトルとの間に，長い text があるのは不適当．)

以上の条件により抽出された候補タグの中から，タイトルをマークアップしていると考えるタグを選択する．その際，HTML タグの特徴と，RSS Feed から収集した約 300 万のタイトルの平均文字長，及び品詞の 3-gram を使用し，以下の手順で選択する．

1, heading タグ (`h1..h6`) である
2, class 名に “title” や “head” などを含む
3, タグがマークアップするすべての text について
- 平均文字長が 20 文字以内
- 収集したタイトルの品詞の 3-gram に適合

以上の特徴を満たすタグがあった場合，それらの中で最も前方に出現するタグをタイトルを示すタグとして選択する．ただし，上記 1,2. の性質を満たすタグがあった場合，それを優先する．また，タイトルが無いと判断された場合は，日付ベースの時系列情報と考える．

2.3 タイトルの自動生成

本節では，タイトルが存在しない，日付ベースの時系列情報にタイトルを付加する手法について述べる．

タイトルの生成に関しては，3 節で述べるように様々な手法がこれまで研究されてきたが，本研究では，extraction ベースの手法でタイトルの付加を行う．この理由は，実際にサービスにする際の処理のスピードを考えてということもあるが，RSS Feed より収集した content:encoded を持つ 100 万件のタイトルと本文のペアを調査した結果，表 1 に示すように，多くの場合でタイトルに必要な語が本文に現れるという特徴を持っていたためである．

よって，本研究では，

- 複数の一連の記事に同時にタイトルを生成
- 記事の書かれた時期に特徴的に現れる単語を重要視
- 大量に収集した RSS Feed から得られる統計情報を利用

する，extraction ベースのタイトルの生成を提案する．タイトルの生成は，“タイトル候補の抽出”，“適切なタイトルの選択” の二段階手法で行う．次節以降で，それぞれの段階について詳説する．

[1] `http://ping.bloggers.jp/`
[2] 収集を開始したのが 12 月の初旬だったため，それ以前の日付を持ったデータは非常に少ない．
[3] `http://dublincore.org/documents/dces/`
[4] `http://web.resource.org/rss/1.0/modules/content/`
[5] 記事の識別番号が，id 属性に付加されているケースがあったため，id 属性については，属性値が異なっている場合でも単一のタグでマークアップされていると考えることにした．

表 1: タイトルと本文の関係

ケース	数	割合
タイトルがそのまま本文に現れる	256,331	(25.6%)
タイトルの内容語がすべて本文に現れる	725,394	(72.5%)
タイトルの内容語が一つ以上本文に現れる	820,042	(82.0%)
タイトルの内容語が一つも本文に現れない	179,958	(18.0%)

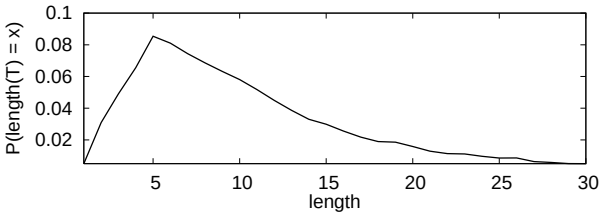

図 4: タイトルの文字長

2.3.1 タイトル候補の抽出

本文からタイトルとして適切な範囲を選択する．使用する素性は，以下の二つである．

1. 文字長 n のタイトルが選択される確率
2. タイトルの先頭，末尾を考慮した品詞 (活用形を含む) の 3-gram に基づく，タイトルの生成確率

これらの確率は，RSS Feed から収集したタイトル 3,281,080 から計算した．図 4 にタイトルの文字長の分布を示す．また，タイトルの品詞 3-gram については，3,177,896 のタイトルから計算を行った．

以上の統計情報を元に，以下の手順で本文中からタイトルの候補を抽出する．

1. 本文を HTML タグで分割し，形態素解析を行う．
2. 名詞，未知語の連続している部分は一つの形態素と考え，品詞は最後の形態素の品詞とする．
3. 形態素解析した文の任意の範囲の形態素列において，以下の条件を満たす形態素列を抽出
   - その形態素中に含まれる，先頭，末尾形態素を考慮したあらゆる 3-gram が，収集したタイトルに出現する
   - 括弧を含む場合は，括弧の対応がとれている

以上の手順によって得られた候補に対して，式 1 によってスコアを計算する．

$$\begin{aligned} score(T) &= log(P(length_{char}(T) = n)) \\ &+ log((\prod_{i=0}^{n+2} P(p_i|p_{i-1}, p_{i-2}))^{\frac{1}{n+3}}) \quad (1) \end{aligned}$$

T は，タイトルの bag-of-words を，右辺の第一項は，文字長 n のタイトルが選択される確率を示す．右辺の第二項で，3-gram の生成確率の相乗平均を計算している理由は，タイトルの長さが長くなるにつれ，この品詞列の生成確率が低下してしまうのを防ぐためである．最終的には，スコアが閾値以上の候補を最終的なタイトル候補として出力する．

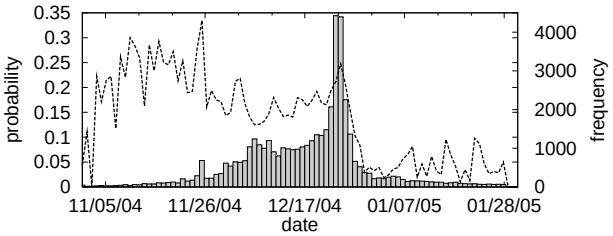

図 5: 「クリスマス」を本文に含む記事数の推移とタイトルへの出現確率の推移

2.3.2 適切なタイトルの選択

以上の処理によって抽出された候補の中から，最も適切なタイトルを選択する．

本研究では，以下の三つの要素を考慮し，適切なタイトルを選択する．

1. $tf(w_i)$: 本文中の形態素 w_i の頻度
2. $idf(w_i)$: 同時にタイトルを付ける複数の本文において，その形態素 w_i を含み，かつタイトルを付けようとしている文書より前に書かれた文書の数の逆数 (タイトルにふさわしい語だったとしても，すべての記事に同じタイトルが付くのは不自然)
3. $P_{Date}(w_i \in T_{Date}|w_i \in D_{Date})$: その単語がある時期に本文に現れた際，それがタイトルに含まれる確率 (例えば，1 月に「クリスマス」という単語が本文に登場しても，タイトルにはなりにくい)

なお，w_i は，タイトル中の i 番目の形態素を示し，T, D は，それぞれタイトル，本文の bag-of-words である．

$P_{Date}(w_i \in T_{Date}|w_i \in D_{Date})$ に関しては，記事の書かれた日付に対して，以下の四つの確率を考える．

- P_{day}: 記事を書いた日付と直前二日間に書かれた記事から計算
- P_{month}: 同じ月に書かれた記事から計算
- P_{year}: 同じ年に書かれた記事から計算
- P_{all}: 収集されたすべての記事から計算

P_{Date} は，P_{day} が定義されていればそれを，定義されていなければ，P_{month} を，さらにそれも定義されていなければ，P_{year} をといったように，順に適用する．なお，これらの確率は，収集している RSS から動的に計算され，1 時間毎に更新される．

図 5 に，2004 年 11 月 1 日から 2004 年 12 月 31 日までの期間において，“クリスマス” を本文に含む記事数の推移と，本文に “クリスマス” が登場したときに，タイトルに “クリスマス” が含まれる確率の推移を示す．図 5 から以下のことが観察できる．

- 12 月中では，“クリスマス” を含む記事数は増えており，12 月 25 日付近で最大である
- クリスマス以降では，記事数は 11 月と同等程度であるにも関わらず，タイトルに含まれる確率は非常に低くなる

- 2005-01-28
  明日はいよいよカザフスタン戦。日本代表の今年の初戦ですね。ドイツワールドカップの最終予選に向けて，良い結果を期待します！
- 2005-01-29
  カザフスタン戦、勝ちました！圧勝でしたね！雨の中，横浜国際総合競技場まで行って良かった！この調子で最終予選も突破して欲しいですね。
- 2005-01-30
  昨日のカザフスタン戦で興奮したせいか、昨日はあんまり眠れませんでした。おかけで大遅刻。

図 6: 記事の例 (1)

表 2: 生成されたタイトル

記事	タイトル	スコア (log)
2005-01-28	カザフスタン戦	-4.09332785321058
2005-01-29	横浜国際総合競技場	-4.48821544478775
2005-01-30	おかけで大遅刻	-5.01575871021318

これらの統計情報を元に，各タイトル候補に対して，式 2 で score を計算し，reranking を行う．

$$score(w_1, ..., w_n) = P_{length}(n) \quad (2)$$
$$* \quad \sum_{i=1}^{n} tf * idf(w_i)^2 * P(w_i \in T_{Date} | w_i \in D_{Date})$$

なお，w_i はタイトル候補中の i 番目の形態素を示す．(ただし，助詞，助動詞，名詞-非自立，動詞-非自立，形容詞-非自立，名詞-代名詞，記号は除いて計算する．)

最終的に score が最も高かったタイトル候補を，その記事に対するタイトルとする．

2.4 適用例

図 6 の 3 つの記事について，以上の処理を適用したときの結果を表 2 に示す．この例では，一つめの記事のタイトルとして選ばれた「カザフスタン戦」がすべての記事に現れるが，二つめ，三つ目の記事では，本文に出現したときにタイトルに出現する確率が高いにもかかわらず，タイトルとして選択されていない．

また，図 7 の例において，日付を変化させたときのタイトル変化を表 3 に示す．このように，記事の日付に対して，適切なタイトルが生成されている．

3 関連研究

本文からのタイトル生成に関しては,，“Headline Generation”，“keyword extraction” といったキーワードで様々な研究が行われている．

本研究と最も関連があると考えられるのは，Banko ら [4] の研究である．Banko らは，Brown ら [5] の提案した統計的機械翻訳手法を用いたタイトル生成手法を提案した．この研究では，記事本文とタイトルの単語間のマッピングと，タイトルの言語モデルを利用した，タイトル生成を行っている．

本研究で提案した手法は，大量の RSS Feed が利用可能であるという特徴を利用し，タイトルと記事のマッピングの部分に，記事の書かれた日時の情報を導入した．これにより，その記事の書かれた記事に適切なタイトルを付加できると考えている．また，ニュース記事と比較して，本研究が対象としている Weblog や BBS などでは，一般的にタイトルも記事本文も非常に短いケースが多く，またトピックも非常に多様であるため，提案手法では extraction ベースの手法を採用した．

お正月と言えば、「あけましておめでとう！」。クリスマスと言えば、「メリークリスマス！」。今年の成人式は、1 月 10 日。

図 7: 記事の例 (2)

表 3: 日付を変化させたときのタイトルの変化

日付	タイトル
2004-12-25	クリスマス
2005-01-01	あけましておめでとう
2005-01-10	今年の成人式

4 おわりに

本論文では，時系列情報を記述する Web ページ中の日付表現とタイトルを検出することにより，RSS Feed を自動的に生成する手法を提案した．また，その際，必須のタスクとなる，タイトルの検出と生成について，大量に収集した Web 上の RSS Feed を利用した手法を提案した．本研究で提案したタイトル生成手法では，記事の書かれた日付を考慮した言語モデルを適用しているため，より適切なタイトルを付加できると考える．

今後の課題は，提案手法の評価である．評価は，(1) 他の手法と比較してよりよいタイトルが付加されているか，(2) 記事の書かれた日付を有効に使えているか，の二点について行う予定である．

また，本研究で提案した RSS Feed の自動生成システムは，今春公開予定である．

参考文献

[1] RSS-DEV Working Group. Rdf site summary (rss) 1.0. `http://web.resource.org/rss/1.0/spec`, 2000.

[2] 神崎正英. Rss – サイト情報の要約と公開. `http://www.kanzaki.com/docs/sw/rss.html`, 2001.

[3] 南野朋之, 鈴木泰裕, 藤木稔明, 奥村学. blog の自動収集と監視. 人工知能学会論文誌, Vol. 19, No. 6, pp. 511–520, 2004.

[4] Michele Banko, Vibhu O. Mittal, and Michael J. Witbrock. Headline generation based on statistical translation. In *the 38th Annual Meeting of the Association for Computational Linguistics*, pp. 218–325, 2000.

[5] Peter F. Brown, Stephen A. Della Pietra, Vincent J. Della Pietra, and Robert L. Mercer. The mathematics of statistical machine translation: Parameter estimation. *Computational Linguistics*, Vol. 2, pp. 263–312, 1993.