2C2-3

企業の公式Webサイトからの手がかり語を用いた基本情報属性抽出

Corporate Profile Information Extraction from Web Sites using Clue Phrases

鶴田 雅信[*1] 増山 繁[*1]
Masanobu Tsuruta Sigeru Masuyama

[*1]豊橋技術科学大学
Toyohashi University of Technology

We propose a method to extract corporate profile information of companies from their Web sites. Our method uses only URLs of target companies' Web site and the clue phrase: "Corporate Profile (written in Japanese)" and extracts information as a set of key-value attributes of corporate profile information.

1. はじめに

現在，多くの企業が公式 Web サイトを開設し，情報を公開している．企業が公式 Web サイトに掲載している情報として，主として「会社概要・沿革」,「IR 情報」，および,「個別の製品・事業の詳細情報」が存在する．これらの企業情報のうち，会社概要・沿革，および，IR 情報などのような企業自体の基本的な情報を，本研究では「企業の基本情報」と呼ぶ．企業の基本情報を自動的に探索，抽出，また，収集することができれば，投資家などの投資判断などにおいて有益だと考えられる．本研究では，企業の基本情報は，属性名と属性値の対で構成される，基本情報属性の集合であると定義する．例えば属性名としては,「代表者名」,「資本金」,「従業員数」などが存在し，その具体的なデータは属性値として扱われる．本研究では，企業の公式 Web サイト集合，および,「会社概要」という手がかり語を用いて，企業の基本情報属性を自動的に抽出する手法を提案する．

企業の基本情報は，RDF など，そのデータの意味を機械が理解できるフォーマットで記述されていないことが多い．そのため，データマイニングなどにおける再利用のためには，人間が記述に対して意味のアノテーションを行うこと，もしくは，このような企業が提供する情報をアルゴリズムを用いて抽出し，再利用可能な形にすることが必要となる．これらの企業情報のうち，製品・事業の詳細情報などは一定のフォーマットで大量に記述されていることが多く，Web ラッパー [Kushmerick 00] などを用いた，アルゴリズムによる自動抽出が可能であると考えられる．しかしながら，企業の基本情報は 1 つの企業のサイトには 1 つしか用意されていないことが多いため，Web ラッパーなどの手法をそのまま適用することはできない．また，異なった企業サイト間において共通，かつ，機械可読なフォーマットが定義されておらず，URL も共通のスキームによって指定されているわけではない．

現在，会社四季報，Wikipedia の特定企業の項目などといった，企業の基本情報属性を人手で収集したリストが存在する．図 1 に，それらのリスト，および，企業の公式サイトに掲載された情報の特徴を示す．会社四季報，および，Wikipedia に記載された情報は，機械的にも扱いやすいが，網羅性，もしくは，信頼性の面で，公式サイトに掲載された情報に劣る部分が存在する．一方，公式サイトに掲載された情報は，前述した通り，異なった企業サイト間においてはフォーマットが統一されておらず，URL も共通のスキームによって指定されているわけではないという問題点がある．しかしながら，この問題点を解決できれば，公式サイトからの情報であるため信頼性が高く，かつ，他の情報源よりも多数の属性を収集することが可能となる．

性質	会社四季報など	Wikipedia (infobox)	企業の公式サイト
信頼性	○(△?)	△(○?)	○
非上場企業の情報	△	○	○
属性の網羅性	△	△	○
フォーマットの統一性	○	○	×
リストの存在する場所の探しやすさ	○	○	△

○：良好, △: 一部は良い, ×: 悪い

図 1: 様々な企業の基本情報リストの特性

2. 手法

提案手法は,「抽出対象となる企業の公式 Web サイトのドメイン名集合」，および,「『会社概要』という手がかり語」を入力とし，対象となる企業の基本情報属性の集合を出力とする手法である．抽出対象となる属性名のリストや，企業の基本情報ページへの URL リストなどは必要としない．ここで，本稿では，Web ページを HTML タグに囲まれた部分をノードとした木構造として扱う．Web ページにおいて，あるノードに含まれる文字列とは，そのノードを根として見た部分木に含まれるテキストノード，および，すべてのノードの alt，title 属性の値を HTML 文書における出現順に結合した文字列のことを指す．また，あるノードに含まれる「語」とは，そのノードに含まれる文字列を形態素解析した結果 [*1] の，形態素 1-gram，および， 2-gram のことを指す．なお，特に指定のない場合，(Web) ページとは BODY タグを根とする部分木を，リンクとは A タグ，もしくは，それを根とする部分木のことを指す．

提案手法は，会社概要ページにおいて，子を 2 つのみ持つ部分木が，基本情報属性を含んでいることが多いという直観に基づいた手法である．例えば，図 2 のような部分木では，属性

連絡先: 鶴田雅信，豊橋技術科学大学大学院電子・情報工学専攻，愛知県豊橋市天伯町雲雀ヶ丘 1-1，TEL: 0532-44-6867，FAX: 0532-44-6873，tsuruta@la.cs.tut.ac.jp

∗1 形態素解析には MeCab 0.97 を用いた．

名，属性値は，あるノードの 2 つの子として表現される．しかしながら，図 3 に示した例のように，属性の構造を意識した構造化がなされていない HTML 文書も多く存在する．図 2 の例における部分木は，基本情報属性の抽出にそのまま用いることが可能である．しかしながら，図 3 の例では不可能である．そこで，提案手法では，図 2 のような，子を 2 つのみ持つノードを根とした部分木を，属性名抽出対象部分木と定義し，このような形の部分木に注目した属性名抽出を行う．属性値抽出では，属性名を示すノードとの相対的な位置関係に基づいた簡易なルールを使用する．

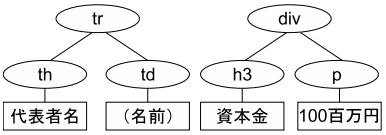

図 2: 子を 2 つのみ持ち，基本情報属性の属性名と属性値がそれぞれ格納された，属性名抽出対象部分木

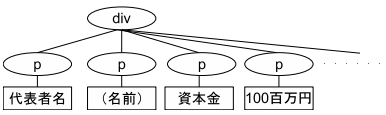

図 3: 属性名抽出対象部分木ではないが，属性を含む例

提案手法では，まず，基本情報ページを探索するために必要な情報を，手がかり語，および，少数の企業の公式 Web サイト全体から収集する (2.1 節で後述)．次に，抽出対象となるサイトをトップページから探索しながら，属性名抽出対象部分木を抽出する (2.2 節で後述)．さらに，抽出された属性名抽出対象部分木から，属性名を抽出する (2.3 節で後述)．最後に，探索の際に辿ったページ，および，Step 3 で抽出された属性名から，属性値を抽出する (2.4 節で後述)．

2.1 企業の基本情報ページベクトルの構築

まず，抽出対象となる企業サイトのうち，少数のものの全体をクロールすることで企業の基本情報が含まれるページに出現する語を素性とする，n 次元の基本情報ページベクトル $\boldsymbol{P_b} = (W_p(w_1), W_p(w_2), ..., W_p(w_n))$ を構築する．$\boldsymbol{P_b}$ の素性値である語 w の重みは，基本情報ページだと考えられるページに含まれ，かつ，それら以外のページにおいてはあまり含まれない語に対して，大きな値が与えられる．$W_p(w)$ は，以下の手順によって求める．

Step A-1 n_l 社の企業の公式 Web サイトに含まれる全てのページをクロールし，収集したものを学習用 Web ページ集合 S_l とする．n_l は定数.

Step A-2 S_l に含まれるすべてのページから，手がかり語が含まれるリンクを抽出し，学習用リンク集合 $L_{positive}$ とする．また，$L_{positive}$ に含まれるリンクの遷移先の Web ページ集合を $P_{positive}$ とする．

Step A-3 $P_{positive}$ に含まれるページ p が属するサイトのトップページから，p への最短路を辿る．そのとき，$P_{positive}$ に含まれるどのページの場合においても経由することのない Web ページの集合を $P_{negative}$ とする．また，$P_{negative}$ に含まれるすべてのページへのリンクを，S_l に含まれるすべてのページから抽出し，学習用リンク集合 $L_{negative}$ とする．また，$P_{negative}$ に含まれる Web ページを遷移先とするリンク集合を $L_{negative}$ とする．

Step A-4 $P_{positive}$ に含まれる Web ページのうち，語 w を含むものの数を $df_{positive}(w)$，$P_{negative}$ に含まれる Web ページのうち，w を含むものの数を $df_{negative}(w)$ とする．

Step A-5 $P_{positive}$ ，および， $P_{negative}$，2 つの Web ページ集合における df 値の偏りに基づいた語 w のスコア $W_p(w) = (df_{positive}(w)/\max_{i=1,...,n}^{w_i} df_{positive}(w_i)) - (df_{negative}(w)/\max_{i=1,...,n}^{w_i} df_{negative}(w_i))$ を，$P_{positive}$ に含まれるすべての語について求める．ここで，$W_p(w)$ が定数 w_{min} より小さい場合，$W_p(w) = 0$ とする．

また，同時に，基本情報ページへの m 次元のリンク文書ベクトル $\boldsymbol{L_b} = (W_l(w_1), W_l(w_2), ..., W_l(w_m))$ の算出も行う．$\boldsymbol{L_b}$ は，基本情報ページへの入次リンクに含まれる語を素性とするベクトルとなる．リンク文書ベクトルにおける語 w の重み $W_l(w)$ は，$W_p(w)$ を求める手順の Step A-4，および，A5 を，学習用リンク集合 $L_{positive}$ ，および，$L_{negative}$ に置き換えた手順によって求める．

2.2 基本情報ページの探索

提案手法では，トップページから基本情報ページらしさの高いリンクを辿ることで探索を行いながら，属性名抽出対象部分木を抽出する．探索の手順を以下に示す．ここで，抽出対象となる企業の数を表す定数を n_e 社とする．

Step B-1 以下のように記号を定義し，初期化する．

属性名抽出対象部分木集合の集合: $Sts = \emptyset$.

探索されたページ集合の集合: $Ps_t = \emptyset$.

企業 ID: $k = 1$.

Step B-2 以下のように記号を定義し，初期化する．

属性名抽出対象部分木の抽出対象となるページ: p_f = 企業 ID が k である企業のトップページ.

探索対象リンク集合: $L_t = \emptyset$.

探索中に最大であった文書スコア: $max_P = 0$.

探索されたページ集合: $P_t^k = \{p_f\}$.

Step B-3 p_f に含まれるリンクをすべて L_t に追加する．また，リンク $\boldsymbol{L} \in L_t$ は，2.1 節と同様に，$\boldsymbol{L_b}$ と同じ素性を持ち，語が出現した場合は 1，しない場合は -1 を素性値とする m 次元のリンク文書ベクトルで表現される．

Step B-4 基本情報ページへのリンク文書ベクトルにもっとも類似したリンク $\hat{\boldsymbol{L}} = \underset{\boldsymbol{L} \in |L_t|}{\operatorname{argmax}} cosine(\boldsymbol{L}, \boldsymbol{L_b})$ を L_t から抽出する．ここで，
$cosine(\boldsymbol{L_1}, \boldsymbol{L_2}) = (\boldsymbol{L_1} \cdot \boldsymbol{L_2})/(|\boldsymbol{L_1}| \times |\boldsymbol{L_2}|)$.

Step B-5 $\hat{\boldsymbol{L}}$ のリンク先であるページを p_{next} とし，p_{next} から文書ベクトル $\boldsymbol{P}$ を求める．$\boldsymbol{P}$ は $\boldsymbol{L}$ と同様に，$\boldsymbol{P_b}$ と同じ素性を持ち，1，もしくは，-1 を素性値とする文書ベクトルである．

Step B-6 文書 p_{next} の基本情報ページらしさのスコア $score_P(\boldsymbol{P}, \boldsymbol{P_b}, \boldsymbol{L}, \boldsymbol{L_b}) = cosine(\boldsymbol{P}, \boldsymbol{P_b}) \times cosine(\boldsymbol{L}, \boldsymbol{L_b})$ を求める．ここで，$max_P < score_P(\boldsymbol{P}, \boldsymbol{P_b}, \boldsymbol{L}, \boldsymbol{L_b})$ であれば $max_P = score_P(\boldsymbol{P}, \boldsymbol{P_b}, \boldsymbol{L}, \boldsymbol{L_b})$ とする．

Step B-7 p_f から属性名抽出対象部分木集合 $St(p_f)$ を抽出し，Sts に加える．

Step B-8 以下の条件のどちらにも合致しなかった場合，P_t^k に p_{next} を加え，$p_f = p_{next}$ とし，Step B-3 に戻る．ここで，doc_{max}，dec_{max} はそれぞれ定数．

- 辿ったページ数が doc_{max} を超える．
- dec_{max} 回連続で max_P が更新されなかった．

Step B-9 Ps_t に P_t^k を加え，$k = k + 1$ とする．

Step B-10 $k > n_e$ であれば Sts ，および， Ps_t を出力し，終了する．$k \leq n_e$ であれば，Step B-2 に戻る．

2.3 属性名抽出対象部分木集合からの属性名抽出

2.2 節における手順の出力のうち，属性名抽出対象部分木集合の集合 Sts を属性名抽出に用いる．また，あるページ p から抽出された属性名抽出対象部分木集合 $St(p)$ に含まれる属性名抽出対象部分木 s を， $s = \langle str_l, str_r \rangle$ という対の形で表記する[*2]．ここで，str_l は属性名抽出対象部分木の根から見て最初の子ノード（左側子ノード）に含まれる文字列，また，str_r は str_l の直後に出現するノード（右側子ノード）に含まれる文字列を指す．Sts から属性名集合を抽出する手順を以下に示す．

Step C-1 Sts において，ある文字列 str_l が左側子ノードに出現する属性名抽出対象部分木を含む集合の数 $df_{left}(str_l)$ を，すべての属性名抽出対象部分木について求める．同様に，Sts において，ある文字列 str_r が右側子ノードに出現する属性名抽出対象部分木を含む集合の数 $df_{right}(str_r)$ を，すべての属性名抽出対象部分木について求める．

Step C-2 ある属性名抽出対象部分木 s における str_l の属性名らしさのスコア $score(s) = df_{left}(str_l)/(df_{right}(str_l) + 1)$ を，すべての属性名抽出対象部分木について求める．

Step C-3 $score(s) > s_{min}$ であった s における str_l すべての集合を属性名集合とする．ここで，s_{min} は定数．

2.4 属性値抽出

属性値抽出は，2.2 節における手順で求められた Web ページ集合の集合 Ps_t に含まれる Web ページ集合 P_t^k に対して行う．

Step D-1 P_t^k に含まれるある Web ページ p から，属性名集合に含まれる属性名と文字列が合致するノードをすべて抽出し，抽出結果を p の属性名ノード集合とする．

Step D-2 ある属性名ノードから次の属性名ノードまでの間に存在するノードを，先行する属性名ノードに対応する属性値ノードとして抽出する．この属性名ノードと属性値ノードの対の集合を，p の属性集合とする．

Step D-3 P_t^k に含まれるすべての p の属性集合の和集合を，企業 ID が k である企業の基本情報属性集合とする．

*2 2 節における定義より，属性名抽出対象部分木は 2 つの子ノードのみを持つ．

3. 評価実験

評価実験では，2009 年 7 月の段階で株式市場に上場していた企業サイトをランダムに選択し，$n_e = 2000$ 社の Web サイトを属性の抽出対象として用いた．また，属性の抽出後に，これらの企業のうち 30 社をランダムに選択したのち，基本情報属性の抽出精度を人手で調査し，評価を行った．さらに，基本情報ページ探索の性能を評価するため，基本情報属性が存在したページを探索経路上で発見できたかどうかについて，成功率を人手で調査した．評価者は，工学系の学生 1 名とした．定数については，$n_l = 100$，$w_{min} = 0.001$，$doc_{max} = 20$，$dec_{max} = 2$，また，$s_{min} = 3$ と設定した．

3.1 実験結果

探索されたすべてのページにおける属性抽出数を表 1 に，その精度 Acc ，および，属性値が一部しか抽出されていなくとも正解とみなした精度 Acc_r を 表 2 に示す．また，正解であった属性を含むページのみに限定し属性抽出を行った場合の結果を表 3 に，その精度を表 4 に示す．なお，正解であった属性を含むページを探索できた企業数は 19 社であった．

表 1: 抽出された属性数

属性の総数 (all)	725
属性値が過不足なく正解であった数 (p_{strict})	512
属性値の一部が正解であった数 (p_{rough})	32
誤って抽出された数	181

表 2: すべてのページにおける精度

$Acc = p_{strict}/all$	0.709
$Acc_r = (p_{strict} + p_{rough})/all$	0.750

表 3: 正解した属性を含むページのみの属性数

属性の総数 (all')	593
属性値が過不足なく正解であった数 (p_{strict})	512
属性値の一部が正解であった数 (p_{rough})	32
誤って抽出された数	49

4. 考察

表 4 より，探索成功時の精度 Acc' は 0.867 ，属性値の一部も正解とする場合の精度 Acc'_r は 0.917 となり，探索の成功時にはよく属性を抽出できていた．しかしながら，表 1 より，探索に失敗した場合には非常に多くのノイズを抽出してしまうという結果が確認できた．探索の失敗例として，画像でリンク探索の手がかりになる情報が記載されているサイトが存在した．また，探索に成功した場合も，PDF によって記述された情報を抽出できない場合が存在した．

その他，属性名の抽出に失敗したため属性値の切り分けに失敗した事例が存在した．図 4 に,「取締役」という属性名によって抽出された 1 つの属性値を示す．この例では，最初の「(人名)」のみが「取締役」に対応する属性値として抽出された場合，正解となる．しかしながら，2，3，および，4 行目の役職名を属性名として抽出することに失敗したため，多くの属性を内包したノード列を属性値として抽出する結果となった．また，3，および，4 行目の属性名は，両方が 5 行目の属性値に対応すべきであるが，提案手法では，属性値は直前の属性名にのみ対応するため，抽出できない属性となる．

表 4: 正解した属性を含むページのみにおける精度

$Acc' = p_{strict}/all'$	0.867
$Acc'_r = (p_{strict} + p_{rough})/all'$	0.917

実行結果	正解	抽出された文字列
属性名 A	属性名 A	取締役
属性値 A-1	属性値 A	(人名)
属性値 A-2	属性名 B-1	○○事業部長
属性値 A-3	属性名 B-2	営業本部副本部長
属性値 A-4	属性値 B	(人名)
属性値 A-5	属性名 C	相談役
属性値 A-6	属性値 C	(人名が続く)

図 4: 属性値切り分けの失敗例

5. 関連研究

教師無し学習の手法を用いて Web から属性名，および，属性値の情報を抽出する手法として，[吉永 07] が存在する．また，[Yoshida 04] は，表における属性名，属性値の典型的なレイアウトの知識を用い，EM アルゴリズムを用いて属性名，属性値を抽出し，それらを集約したデータベースを作成する手法を提案している．これらの研究は本研究と近い目的を持っているため，手法の比較実験など，詳細な検討が必要であると考えられる．

また，[中根 08] では，ブートストラップ的手法を用いて生成したテンプレートを用い，Web 全体からデータベースのスキーマを抽出するための手法を提案しようとしている．提案手法は企業の公式 Web サイトのみを対象とし，企業それぞれの基本情報属性の抽出を目的としているため，直接の比較はできない．[板井 02] は，授業内容のシラバスである HTML 文書を収集し，SVM によって付与したラベル系列に対して，オートマトンを用いて属性名，および，属性値を抽出する研究を行っている．提案手法は人手で訓練データを作成する必要はないが，属性値の抽出手法は類似している．

抽出対象ページの探索については，フォーカストクローリングの分野に先行研究がいくつか存在する [Ester 03, Menczer 04]．提案手法の基本情報ページ探索の正解率は低く，これらの知見をさらに導入する必要があると考えられる．

5.1 表のデータからの情報抽出

提案手法は，現在の段階では，複雑な表のデータに対しての解析を行うことはできない．また，本研究では複雑な表の解析を行っていないため，表を 1 つの大きな属性値として誤って取得してしまう場合があった．2.4 節で解説した手法では，1 行に属性名，属性値の両方が含まれていなければ，正しく情報を抽出することはできない．しかしながら，表データの解析については多くの既存手法が存在するため，これらを利用することで，属性の抽出数を増加させることができると考えられる．例えば，[増田 03] は，HTML の 表形式データ中に隣りあって存在する項目同士の言語的な類似度を定義し，類似度が小さくなる部分に内容的な切れ目が存在するという仮定から，表の構造認識を行っている．前述した通り，[Yoshida 04] は，提案手法と類似した考えに基づいた，表からの情報抽出手法を用いている．また，[大西 06] では，表形式データにおける，複数行，および，複数列にまたがって表示される項目を手がかりに，表の属性名部分，および，属性値部分の分割を行っている．[北山 06] は，人手で与えた正解データから単語の属性名らしさによる重みを算出し，これを用いて表の構造認識を行う手法を提案している．提案手法では，単語の属性名らしさの重みを教師無しの手法で算出することに成功しているため，[北山 06] に類似した手法を用いることでも，複雑な表に対する情報抽出が可能になると考えられる．

6. まとめ

手がかり語「会社概要」，および，企業の公式ページのトップページ URL 一覧を入力とし，企業の基本情報を属性名，属性値の組の形で抽出する手法を提案し，評価実験の結果，ページの探索に成功した企業においてはよい精度を得た．網羅性の高い属性名抽出，および，属性名抽出に失敗した場合にも頑健な手法の考案が今後の課題となる．

謝辞

本研究の一部は，日本学術振興会科研 (C22500129)，電気通信普及財団，人工知能研究振興財団の支援を得て行われた．

参考文献

[Ester 03] Martin Ester, Hans-Peter Kriegel, Matthias Schubert, Accurate and Efficient Crawling for Relevant Websites, Proc. VLDB 2004, pp.396–407, 2004.

[板井 02] 板井 久美, 高須 淳宏, 安達 淳, HTML からの情報抽出と統合, NII journal, Vol. 6, pp.9–19, 2002.

[北山 06] 北山 翼, 嶋田 和孝, 遠藤 勉, 単語の属性名らしさを利用した Web 上の表の構造認識, FIT 2006, pp. 141–143, 2006.

[Kushmerick 00] Nicholas Kushmerick, Wrapper Induction: Efficiency and Expressiveness, Artificial Intelligence, Vol. 118, 2000.

[増田 03] 増田 英孝, 塚本 修一, 安富 大輔, 中川 裕志, HTML の表形式データの構造認識と携帯端末表示への応用, 情報処理学会論文誌. データベース, Vol. 44, No. 12, pp.23–32, 2003.

[Menczer 04] F. Menczer, G. Pant, and P. Srinivasan, Topical web crawlers: Evaluating adaptive algorithms, ACM Trans. Internet Technol., Vol. 4, No. 4, pp378–419, 2004.

[中根 08] 中根 史敬, 大坪 正典, 土方 嘉徳, 西田 正吾, Web からのスキーマ抽出に関する基礎検討, DEWS 2008, 2008.

[大西 06] 大西香織, 田島敬史, Web 上の表データの論理構造の発見, DEWS2006, 2006.

[Yoshida 04] M. Yoshida, K. Torisawa and J. Tsujii, Integrating Tables on the World Wide Web, JSAI Trans., Vol. 19, No. 6, pp.548-560, 2004.

[吉永 07] 吉永直樹, 鳥澤健太郎, Web からの具体物の属性・属性値情報の自動獲得, 言語処理学会第 13 回年次大会発表論文集, pp.887–890, 2007.