類似度に基づく訓練データの獲得とスペック情報の抽出

嶋田 和孝　　遠藤 勉
九州工業大学 情報工学部 知能情報工学科

1　はじめに

近年の Wrold Wide Web (WWW) の急速な普及により，世界中から発信された膨大な電子化文書へのアクセスが可能になった．しかしながら，そのような膨大な情報源から，必要な情報のみを的確に得ることは困難を極める．的確な情報を得るために，テキストを対象とした文書分類や情報の抽出などの様々な技術が注目され，研究されている．しかしながら， Web 上に存在するのはテキスト情報だけではなく，表や画像など様々な表現形式が使用されている．ここで，表形式で記述された情報について着目する．従来の情報検索システムなどでは，表はテキストとして扱われることが多かった．表は属性と属性値によって構造化された情報であり，その特性を考えると，表をテキストとして扱うのではなく，テキスト部分と切り離し，表として認識し，利用することが情報検索システムなどの精度向上に繋がる．また表は情報間の関係を記述するのに適した表現形式であり， Web 上に存在する文書から表を抽出することは， Web Mining や質疑応答システム，要約処理などのための重要なタスクの一つである [1, 2, 4, 6, 8].

本稿では，電子化された情報の一つである，製品のスペック情報の抽出について議論する．一般に，パソコンやデジタルカメラ，プリンタなどの製品の機能や装備などのスペック情報は表形式で記述される．本稿ではこれらの表形式で記述されたスペック情報を性能表と呼ぶことにする．その例を図 1に示す．性能表を扱う理由としては，

- ポータルサイトの存在
  現在， Web 上には，数多くの製品情報に関するポータルサイトやオンラインショッピングサイトが存在する[1]．これらのサイトで，ユーザが製品を比較する際に最も重要な情報の一つが性能表である．多くの製品は頻繁に最新機種が発表され，その度に性能表を人手で収集するのはコストがかかる．膨大な Web ページの中から製品のスペック情報を的確に抽出することは，そのようなポータルサイトの自動構築のために大きな意義を持つ．

- 製品情報のデータベース化
  性能表は表形式で記述されているので，表領域が正しく特定されれば，属性と属性値の切り分けや対応付けなどの解析が比較的容易で，製品データベースの自動獲得が可能になる．これらのデータを利用し，ユーザの要求に合致した製品を選択するシステムなどの構築が可能になる [5].

などが挙げられる．

[1] 価格.com (http://www.kakaku.com/) や Yahoo! Shopping (http://shopping.yahoo.co.jp/) など．

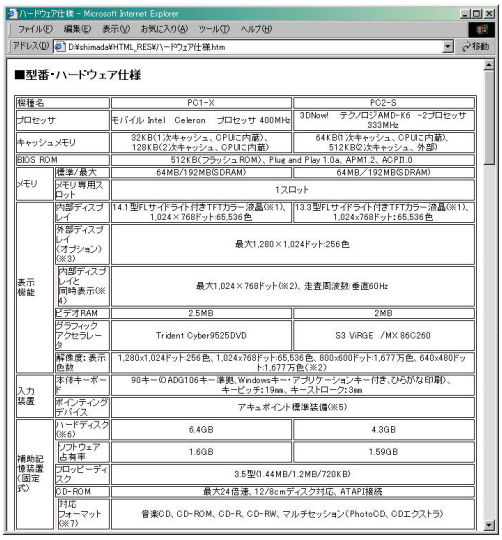

図 1: パソコンの性能表の例

本稿では， Web 上に存在する文書を性能表を含んでいる文書と含んでいない文書に分類するタスクを考える．このタスクにおいて，分類を行う場合に問題となるのが，データ中の正例 (性能表を含んでいる文書) の数である．Web 上に存在する膨大な文書のうち，求めている文書，すなわち正例の数は極端に少ないことが多い．本タスクのための実験データを収集した結果，正例は全体の 2 〜 5% 程度しか存在しないことが確認されている [9]．そのため，少ない訓練データで分類器を作成した場合，訓練データ中に十分な正例が存在しないことが原因で，適切な分類器が作成されないことがある．本稿では，この問題を解決するために，訓練データ中の正例を基にラベル無しデータから新たな正例を獲得する手法を提案する．得られたデータと既存の訓練データを統合し，分類器を生成することで，より精度の高い分類を目指す．

提案手法の処理の流れを図 2に示す．まず，訓練データから任意の正例を 1 つ抽出する．その正例とラベル無しデータ中の文書の類似度を測定する．閾値以上の類似度を持つ文書に対して投票を行い，その文書とさらに類似した文書が存在しないかをチェックする．類似した文書が存在すれば，同様にその文書に投票をする．この処理を繰り返し，最終的にある閾値以上投票された文書を正例だと判断し，その文書群を利用して，分類器を作成する．その分類器を使用して，既存の訓練データを仮の評価データと見立てて評価する．類似度に関する閾値を下げながら同様の処理を行い，新たな訓練データを獲得する．

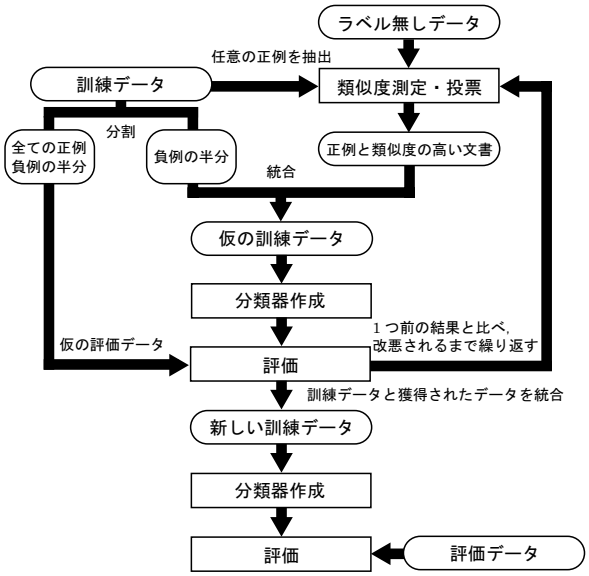

図 2: 処理の概要

2 分類器と素性選択

本節では，使用する分類器とその分類器に用いる素性の選択方法について述べる．分類器には Support Vector Machine (SVM) および Transductive SVM を利用する．素性選択には，正規化した $tf \cdot idf$ を用いる．

2.1 Support Vector Machines

SVM は Vapnik らが考案した Optimal Separating Hyperplane を起源とする，超平面による特徴空間の分割法であり，現在，二値分類問題を解決するための最も優秀な学習モデルの一つとして知られている [7]． SVM は訓練サンプル集合からマージン最大化と呼ばれる戦略を用いて，線形識別関数

$$f(\boldsymbol{x}) = \boldsymbol{w} \cdot \boldsymbol{x} + b \qquad (1)$$

のパラメータを学習する．ここで， $\boldsymbol{x}$は入力ベクトルである．$\boldsymbol{w}$ と b がマージン最大化戦略の際に学習されるパラメータであり， $f(\boldsymbol{x}) \in \{+1, -1\}$ となる．

2.2 Transductive SVM

Vapnik [7] が提案した理論を基に Joachims [3] によって具体化された Transductive SVM (TSVM) は， Transductive 法と呼ばれる，与えられたラベル無しデータの分布に注目し，ラベル無しデータの誤分類の最小化を目的とする学習方法を SVM に適用し，拡張したもので，学習時にラベル無しデータの分布を考慮する事で分類精度を上げる手法である．この手法により，少量の訓練データで高精度の分類器を生成することが可能になる．以下に TSVM のアルゴリズムを示す．

Step 1 訓練データを基に SVM で分類器を生成する．

Step 2 得られた分類器を用いてラベル無しデータを分類する．得られた分類結果をそれぞれのラベル無しデータの仮クラスとする．

Step 3 仮クラスの付与されたラベル無しデータを訓練データに含め， SVM によって分類器を再生成する．

Step 4 マージン内のラベル無しデータのうち，各々の仮クラスを入れ替えることでマージンを最大化できるペアを見つけ，入れ換える．入れ換えられたデータセットを用いて， SVM による再学習を行う．この処理の際に，ラベル無しデータ中の正例および負例の分布を考慮する[2]．

Step 5 入れ換えるペアがなくなるまで **Step 4** を繰り返す．

2.3 素性選択

続いて， SVM および TSVM のための素性選択について述べる．本研究では，以下の条件を全て満たすものを素性候補とした．

(1) 表の属性欄中に出現する単語

(2) 一定長以内の文章中に出現する単語

(3) 性能表が存在する文書および性能表が存在しない文書内で顕著または限定的に出現する単語

これらの条件に基づき，素性となる候補を Web 文書から抽出する．条件 **(1)** では表中の要素を属性および属性値に切り分ける必要がある．ここでは，一般に殆どの性能表は第 1 列目 (最左列) に属性が現れ，それより右側の列に属性値が存在するという経験則から，最左列の要素を属性だと解釈する．素性候補の抽出は，以下の手順で行われる．

1. HTML 文書から<TABLE> タグで記述された領域を抽出する．
2. <TABLE> タグ中の各<TR> タグ中の初めの<TD> タグの内容を抽出する．
3. 得られた文字列が 25 文字以内であれば，形態素解析[3]を行い，素性候補を抽出する． 25 文字という制約は経験的に定められた．

素性選択には， Wang ら [8] が表抽出処理に拡張し，正規化した $tf \cdot idf$ 法を用いる．ここで，本研究では，素性候補の抽出条件を考慮する．すなわち，素性の候補となる単語 t としては，文書 d 中の<TABLE> タグにおける最左列の単語のみを利用する．さらに，これを学習用に拡張し， $D = \{D_{real}, D_{no}\}$ とする．ここで， D_{real} は性能表を含む文書群， D_{no} は求めている製品の性能表を含まないもしくは性能表以外のテーブルを含む文書

[2]一般には，ラベル無しデータ中の正例および負例の分布比率は未知なため，訓練データ中の正例と負例の比率などを参考にして求められた予測比率を利用し，パラメータが調整されることが多い．

[3]形態素解析には奈良先端科学技術大学で開発された「茶筌」を用いた．http://chasen.naist.jp/hiki/ChaSen/

群である．具体的には，以下の式で各語の重みが算出される．

$$w_t^{real} = \sum_{d_i \in D_{real}} tf(t,d_i) \times \log(\frac{df_t^{real}}{N_{real}} \frac{N_{no}}{df_t^{no}} + 1) \quad (2)$$

$$w_t^{no} = \sum_{d_i \in D_{no}} tf(t,d_i) \times \log(\frac{df_t^{no}}{N_{no}} \frac{N_{real}}{df_t^{real}} + 1) \quad (3)$$

ここで，df_t^{real}, df_t^{no} は D_{real} および D_{no} における単語 t の df 値である．また，N_{real} および N_{no} は，D_{real} および D_{no} に属する文書の総数を表す．最終的な重みは以下の式で求める．

$$ws_t^{real} = \frac{w_t^{real}}{Norm_{real}}, \quad ws_t^{no} = \frac{w_t^{no}}{Norm_{no}} \quad (4)$$

ただし，

$$Norm_{real} = \sqrt{\sum_{t \in D_{real}} w_t^{real} \times w_t^{real}} \quad (5)$$

$$Norm_{no} = \sqrt{\sum_{t \in D_{no}} w_t^{no} \times w_t^{no}} \quad (6)$$

ここで閾値以上の値を持つ ws_t^{real} および ws_t^{no} を類似度計算や分類器作成のための素性として扱う．

3 類似度に基づく訓練データの獲得

現在設定しているタスクでは，サンプリングした訓練データ数が少ない場合に，正例が極端に少ないという問題がある．実験に使用したデータの場合，正例の全体に占める割合は，2 ～ 4% 程度である．訓練データ中の正例が極端に少ない場合，それらを使用して得られた分類器の精度も悪くなる．そこで，ラベル無しデータから，新たな正例を獲得する．

提案手法では，正例同士は類似しているという仮定に基づき，訓練データ中の任意の正例とラベル無しデータの類似度を測定することで，ラベル無しデータから新たな正例を獲得する．具体的な処理の流れを以下に示す．

(1) 訓練データから正例を 1 つ抽出する．

(2) その正例とラベル無しデータ中の文書の類似度を測定する．

(3) 閾値 th_1 以上の類似度を持つ文書に投票する．

閾値 th_1 以上の類似度を持つ文書群に対して

(3-1) 閾値 th_1 以上の文書それぞれとラベル無しデータ中の文書の類似度を測定する．

(3-2) 閾値 th_1 以上の文書が存在すればその文書に投票する．

(4) 投票数が閾値 th_2 以上の文書を正例として抽出する．

表 1: データセット

	パソコン	デジタルカメラ	プリンタ
性能表を含んでいる文書数	239	117	143
性能表を含んでいない文書数	4761	4883	4857

(5) 訓練データを (a) 正例の全てと負例の半分，(b) 負例の半分，に分割する[4]．

(6) **(4)** で得られた文書と (b) のデータを仮の訓練データとして分類器を作成する．

(7) 得られた分類器で (a) を評価する．

(8) th_1 を適度に下げ，**(2)** から繰り返す．

(9) 各繰り返しごとの **(7)** の結果を比較し，その精度が改悪される場所を探す．

(10) 改悪される直前に獲得された文書を正例とみなして訓練データ中に追加する．

上記の手法で獲得された新たな訓練データを用いて分類器を作成し，評価データを分類・評価する．**(10)** において，改悪されたかどうかの基準はいくつか考えられるが，ここでは，適合率が低下する場合を「改悪された」と定義する．これは，**(7)** の評価において適合率が低下するということは，負例を誤って正例として抽出した可能性が高いと考えられるためである．

4 実験

本節では，評価実験について述べる．実験対象となる製品は，パソコン，デジタルカメラおよびプリンタの 3 種類とした．実験データは，実際の Web から得られた文書群からランダムに抽出された 5000 文書を使用した．5000 文書からランダムに抽出した 100 文書を訓練データとし，残りの 4900 文書を評価データとした．評価データのうち，1000 文書をラベル無しデータとして提案手法に利用した．データ中の正例と負例の内訳を表 1に示す．但し，性能表を含んでいない文書群中には，別の製品の性能表が含まれている．例えば，デジタルカメラの場合は，性能表を含んでいない文書にフィルムカメラやビデオカメラなどの性能表が含まれていることがある．

評価には以下の尺度を用いた．

$$\text{適合率(P)} = \frac{\text{抽出された文書中で性能表を含んでいる文書の数}}{\text{抽出された文書の総数}} \quad (7)$$

$$\text{再現率(R)} = \frac{\text{抽出された文書中で性能表を含んでいる文書の数}}{\text{性能表を含んでいる文書の総数}} \quad (8)$$

$$F\text{値}(F) = \frac{1}{\frac{1}{2P} + \frac{1}{2R}} \quad (9)$$

提案手法で用いる分類器には SVM[5] を利用した．提案手法の th_1 は 0.95 を初期値とし，0.05 ずつ下げて，0.35

[4] この処理は初めの 1 回のみ．すなわち，繰り返し中で (a) および (b) は変化しない

[5] SVM^{light}: http://svmlight.joachims.org/

表 2: 実験結果

製品	評価尺度	(1) SVM	(2) 提案手法	(3) TSVM	(4) 提案手法 + TSVM
パソコン	適合率	**0.9171**	0.8438	0.8384	0.8360
	再現率	0.6928	0.8662	0.8789	**0.9112**
	F 値	0.7711	0.8363	0.8551	**0.8708**
デジカメ	適合率	**0.9582**	0.8613	0.7264	0.7218
	再現率	0.5201	0.8428	0.8348	**0.8840**
	F 値	0.6408	**0.8391**	0.7608	0.7762
プリンタ	適合率	**0.8944**	0.7971	0.7670	0.8402
	再現率	0.4798	**0.8612**	0.7891	0.8266
	F 値	0.5734	**0.8247**	0.7574	0.8179

まで繰り返した．類似度の測定には，*cos* 尺度を用いた．th_2 は訓練データから抽出された正例と類似度が th_1 以上だと判断された文書数の 2/3 とした．最終的な評価データの分類実験では，(1) SVM，(2) 提案手法を基に獲得した訓練データを用いた場合の SVM，(3) TSVM，(4) 提案手法を基に獲得した訓練データを用いた場合の TSVM，の 4 つを比較した．提案手法と TSVM で使用したラベル無しデータは同じものである．実験結果を表 2に示す．表中の太字は各評価尺度での最高値を表す．

パソコンの場合は，提案手法と TSVM を組み合わせたものが，デジタルカメラとプリンタの場合は提案手法が最も高い F 値を得た．提案手法および TSVM に共通するのは，SVM に比べ，再現率が大きく上昇することである．これは，提案手法では訓練データの獲得処理が，TSVM の場合は Transductive 法による学習が有効に機能していることを表している．適合率は，どの場合においても SVM が最も高くなるが，これは他の手法に比べ，再現率が極端に低いためである．この実験結果より，提案手法の有効性が確認された．

今後の課題としては，提案手法における，最適な閾値 (th_1) の推定が挙げられる．実際，現在の手法では，最適な閾値だとされた場合においても，負例を正例だと判断し，訓練データと統合する場合がある．より精度を向上させるには，最適な閾値をいかに自動的に推定するかが課題となる．

5 おわりに

本稿では，製品のスペックを記述した表を含む文書の抽出処理について述べた．サンプリングされた訓練データが少ない場合，負例に対して正例が極端に少ないという問題点の解決方法について提案した．実験の結果，通常の SVM に比べ，提案手法もしくは提案手法と TSVM を組み合わせた手法が高い F 値を得た．今後の課題としては，提案手法における閾値の推定法についての改良などが挙げられる．

参考文献

[1] M. Hurst. Layout and language: Challenges for table understanding on the web. In *Proceedings of Workshop on Web Document Analysis, WDA01*, pp. 27–30, 2001.

[2] K. Itai, A. Takasu, and J. Adachi. Information extraction from html pages and its integration. In *Proceedings of the 2003 Symposium on Application and the Internet Workshops (SAINT03)*, pp. 276–281, 2003.

[3] T. Joachims. Transductive inference for text classification using support vecor machines. In *Proceedings of the Sixteenth International Conference on Machine Learning*, pp. 200–209, 1999.

[4] D. Pinto, A. McCallum, X. Wei, and W. B. Croft. Table extraction using conditional random feilds. In *Proceedings of the 26th annual international ACM SIGIR conference on Research and development in informaion retrieval*, pp. 235–242, 2003.

[5] K. Shimada and T. Endo. Product specifications summarization and product ranking system using user's requests. In *Information Modelling and Knowledge Bases XV, IOS Press*, pp. 315–331, 2003.

[6] K. Shimada, T. Ito, and T. Endo. Multiform summarization from product specifications. In *Proceedings of PACLING 2003*, pp. 83–92, 2003.

[7] V. N. Vapnik. *Statistical Learning Theory*. Wiley, 1999.

[8] Y. Wang and J. Hu. A machine learning based approach for table detection on the web. In *Proceedings of The Eleventh International World Web Conference*, 2002.

[9] 林晃司, 嶋田和孝, 遠藤勉. 機械学習を用いた www からの製品性能表の分類と抽出. 言語処理学会第 10 回年次大会, pp. 733–736, 2004.