社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2003－DBS－130 (22)
2003－ＦＩ－ 71 (22)
2003／5／23

複数尺度を用いた参考文献の同定

伊藤 敬彦† 堀部 史郎†‡ 新保 仁† 松本 裕治†

ある文献が他のどの文献を参照しているか、という文献の参照情報は、その文献の参考文献一覧の各一文（参考文献文）が指し示している文献を、文献データベース（著者、題目、掲載誌等からなる文献を表す文献データ集合）中から同定することで獲得できる。この同定を、参考文献文と文献データの単なる文字列の完全一致判定で行なうことはできない。参考文献文に表記の多様性や誤りが存在するためである。
本稿では参照情報の自動獲得手法を提案する。始めに、単一のベクトル空間とその上での類似度を用いて粗く候補を絞る。次に、参考文献と文候補が同一の文献であるかを多数の尺度に基づく類似度を特徴量として判定する。複数の尺度それぞれの重みを人手でつけることは現実的ではないため本稿ではサポートベクターマシーンを用い、各尺度の最適な重みを自動で算出した。結果 F 値 0.992 が得られた。

Citation Indexing Using Many Similarity Measures

Takahiko Itoh† Shiro Horibe†‡ Masashi Shimbo† Yuji Matsumoto†

Citation indices are invaluable for the retrieval of related papers. With the increase in the volume of scientific literature, a demand is growing for methods to automatically construct such indices. However, a naive method such as using exact string matches makes errors because of the various ways references can be formatted. In this paper, we propose a new citation indexing method that uses many features to evaluate similarity between references and bibligoraphic data. Unlike the previous work which typically uses only a few features to compute similarity, our method computes a weighted sum of more than 1200 feature values, each of which reflects one of the diverse similarity measures. An F-measure of 0.992 was obtained when Support Vector Machines were used to compute optimal weights to each feature.

1 はじめに

近年、文献の参照情報の利用が進んでいる。文献の参照情報とは、ある文献が他のどの文献を参照しているか、または他のどの文献から参照されているか、という情報のことである。この情報は、文献検索システムの出力として、あるいは、個々の文献の価値の算出にも使われている。

ある論文が参照している論文は参考文献一覧の各一文（以下、参考文献文と呼ぶ）に記載されている。そのため、文献の参照情報は参考文献文文字列の著者、掲載誌、出版年月日などを含む情報を手がかりに同定することができる。この処理を人手で行なうことも可能であるが、大変な労力が必要になる。

そこで、現在、参考文献文から参考文献を同定する処理の自動化が望まれている。この情報を自動獲得する問題は、電子化された参考文献文集合と文献データベース（文献を表すデータ集合、以下この個々のデータを文献データと呼ぶ）が存在すれば、参考文献文を文献データベースから検索する問題となるが、この同定を、参考文献文と文献データの単なる文字列の完全一致判定で行なうことはできない。その理由の一つは、参考文献文には題目、著者名、掲載誌等の情報を含み、それらの書かれる順番や表記にも多様性が存在することである。また、参考文献文と文献データベースの双方に誤りが含まれる可能性がある。誤りの原因には引用者によるものと、対象とする参考文献文の文字列情報がOCR で読み取ったデータであれば、読み取り誤りによるものもある。さらに、省略表現が多用されるという、参考文献文の特徴もこの判定処理を困難なタスクにしている。

今回、我々は OCR で読み取られた参考文献文を対象とし、文献の参照情報の自動獲得を試みた。

我々の提案する手法の特徴は多数の異なった尺度に基づいて、類似度を計算する点である。参考文献文と文献データ（著者、題目、掲載誌等のフィールドを持つ）間のこれら類似度の重みつき和がある閾値以上であれば、それらは一致したと判定する。

†奈良先端科学技術大学院大学 情報科学研究科
Graduate School of Information Science, Nara Institute of Science and Technology
{takahi-i,shiro-ho,shimbo,matsu}@is.aist-nara.ac.jp
‡現リコー (株)

なお、個々の類似度に対する重みは教師付学習法により自動的に算出する。

ただし、参考文献文と文献データの数が非常に多いことから、まず始めに軽い処理で類似度を計算し、候補を絞り込んだ。この後、提案する複数尺度を用いた同定処理を行なった。

本稿の構成は以下の通りである。次の節で、関連研究を紹介し、その後、我々の提案手法の全体像、使用する尺度の内訳と実験結果、そして考察を述べる。

2 背景

WWW 上にある文献を対象に、参照情報の自動獲得を目指した研究は幾つか存在する。

Citeseer [4] は、英語の論文を対象とした大規模な論文検索システムである。Lawrence ら [3] は、参考文献文の集合を同一の文献を指す参考文献文同士にクラスタリングする、というタスクにおいて複数の手法を比較している。単語の 1-gram と 2-gram を特徴量とする二値ベクトル表現を用いて、共通する単語（の組）の数と共通しない単語（の組）の数の比を利用する手法の方が、編集距離の一種である LikeIt [1] より、良い結果が得られると報告している。Citeseer のシステムにおいては、さらに、参考文献文の著者、題目、出典、日付などの各フィールドをヒューリスティックな手法で特定し、その各フィールド毎に単語の二値ベクトルを用いた手法でクラスタリングしており、実験として、1158 件の参考文献文をクラスタリングした結果、5 パーセントのクラスにのみ誤りが含まれていたと報告している。そして、そのようにして得られた各クラスを用いて参照情報を獲得している。

Cora [5] も、英語の論文を対象とた論文検索システムである。基本的に Citeseer と同じ手法を用いて参照情報を獲得しているが、参考文献文の各フィールドをヒューリスティックな手法ではなく、隠れマルコフモデル [6] を用いて特定するという点で異なる。実験として、200 件の参考文献文の 4479 単語に、フィールドのラベルを付けるというタスクにおいて、90 パーセント以上の性能が得られたと報告している。

難波ら [8] は日英の文献を対象として、参照情報を獲得している。参考文献文同士をクラスタリングしたり、フィールド毎に分割したりせずに、個々の参考文献文と文献データから、文字の 6-gram を 2 文字ずつずらして取り出し、これらがある閾値以上一致するかどうかで同定している。

OCR の読み取りデータから、参照情報を獲得しているデータベースも存在する。ISI 社の Science Citation Index は、自然科学分野を対象とした文献データベースである。また、同社の Journal Citation Reports では、学術雑誌の比較、評価を行なっている。ある雑誌の掲載文献への引用が多ければ多いほど、その雑誌は権威がある、という考えに基づき参照情報の利用がなされている。

ISI 社は人手も用いて参照情報を獲得しているが、その自動化の試みとして、Hitchcock [2] らの Open Journal project がある。ここでは、参考文献文の各フィールドがすでに特定されたデータを対象としており、まず参考文献文の著者名フィールドと掲載年フィールドを用いて一致する文献データを絞り込んでいる。その後、題目のフィールドに含まれる誤りへの対応としていくつかの前処理を行い、共通しない単語の数を閾値に用いて同定している。

また、三平ら [10] は、日本語を対象とし、参考文献文と文献データの同定に注目している。参考文献文の一部のフィールドをヒューリスティックな手法を用いて特定し、それらのフィールドについて一致する文献データに候補を絞り込んでいる。その後、候補となった文献データの題目と参考文献文から文字 2-gram を取り出し、一致する文字組の数と参考文献文の文字組の数との比を基に、文献データ候補をクラスタリングして同定している。実験として、59 件の参考文献を同定するという比較的小規模なタスクにおいて、58 件の参考文献を正しく同定できたと報告している。

WWW 上の文献を対象としているシステムには、網羅的な文献の収集が困難だという問題点がある。その点、OCR 読み取りを対象とした場合、データベースに加える文献は恣意的に決められるため、システムの用途に応じて電子化されていない古い文献を加えることや、ある特定分野の雑誌に関しては全て網羅するといったことも可能である。

3 文献同定処理の全体像

本研究は、文献の参照情報を自動獲得することを目的とし、OCR で読みとられた参考文献文を対象とした。具体的には、文献データベースと OCR 読み取りされた参考文献文集合を入力とし、参考文献文の各文が指し示す文献が文献データベース中に存在するか、存在するとすればどの文献データであるかを同定した。

提案手法の特徴は参考文献文と文献データの文字列の間の類似度を様々な尺度で計測することである。複数尺度を用いる手法は計算負荷が大きいため、前処理として参考文献文の候補を文献データベースから軽い処理で荒く絞り込んだ。

提案する複数尺度を用いた同定手法と絞り込み処理は、いずれもベクトル空間モデルに基づいている。ベクトル空間モデルとは、対象を特定のベクトル空間に写像して、一つのベクトルとして扱う手法のことである。文字列をベクトル空間に写像するこ

とで、文字列同士の類似度をベクトル同士の類似度として扱うことが可能になる。

3.1 絞り込み処理

今回は先行研究 [9] の結果を基に、以下のような絞り込み手法をとった。始めに、参考文献文と文献データを単語 1-gram を基底に持つ二値ベクトル空間に写像した。

全ての参考文献文ベクトルと文献データベクトルからなるペアについて 3 つの特徴量 (v, r, m) を計算し、それぞれに閾値を決定する。これらの閾値の全てを満足する参考文献文と文献データのペアのみを、候補として次の複数尺度を用いた同定処理にまわした。

以下具体的に述べる。入力は C (参考文献文集合) と D (文献データ集合) である。そして C と D の全ての組合せ $s_i = \langle c_i, d_i \rangle (c_i \in C, d_i \in D)$ について特徴量 (v_i, m_i, r_i) を計算する。v_i は $\max((\vec{c_i} \cdot \vec{d_i}/|\vec{c_i}^2|), (\vec{c_i} \cdot \vec{d_i}/|\vec{d_i}^2|))$ とする。ただし $\vec{c_i}$,$\vec{d_i}$ は c_i,d_i の単語 1-gram を基底に持つベクトル表現である。この特徴量 v_i の意味については 4.2 節で詳しく述べる。特徴量 m_i は $v_i^{\max} - v_i$ で与えられる。ここで $v_i^{\max} = \max_j \{v_j \mid c_j = c_i\}$ 、すなわち $v_i^{\max}$ は参考文献文 c_i と全文献データとの類似度の最大値である。なお、この特徴量 m_i は 4.4 節で述べる尺度「最大値との差」と同じである。

r_i は参考文献文 c_i と全文献データとの類似度の順位で与えられる。

なお、絞り込みの際に用いるこれら 3 つの特徴量の閾値の決定方法については後述する（5.1 節）。

3.2 複数尺度を用いた同定

本節では、絞り込み処理の次の段階として、複数の尺度を用いた参考文献の同定を行う。3.1 節の絞り込み処理で生成された事例についてそれを構成する、参考文献文と文献データ文字列間の類似度を複数の尺度で計り、それぞれの尺度におけるペアの類似度に重み（各尺度の重要度を表す）を乗じて足し合わせる。この値が閾値を超えれば、その参考文献文に対し文献データは正解文献データであると判定し、さもなくば、不正解とみなす。

3.1 節で取り扱った絞り込み処理では、参考文献文と文献データとの類似度の尺度として 1 種類のみを使用した。これは、絞り込み処理では大量のデータを対象とするので、軽い計算処理で測れる尺度が望ましいためである。しかし、一旦、正解文献データの候補に絞り込んだ後ならば、同定するための尺度の選択において、計算処理の重さにさほどこだわる必要はない。そこで多数の尺度を用いる。実際に使用する尺度 1248 種類の内訳については 4 節で説明し、本節では、それらをどのように用いるかについて述べる。

各尺度における類似度はそのまま足し合わせるのではなく、尺度の重みとして分類に有効な尺度に大きな値を掛け、さほど有効ではない尺度に小さな値を掛けて足し合わせる。参考文献文 c と候補文献データ d が与えられ、それらの n 個の類似度の値を $\vec{x} \in \mathbb{R}^n$、各尺度の重みを $\vec{w} \in \mathbb{R}^n$、閾値を $b \in \mathbb{R}$ であるとき、判別関数は

$$g(\vec{x}) = \mathrm{sgn}(\vec{w}^T \vec{x} + b) \tag{1}$$

で表せる。つまり、$g(\vec{x}) = +1$ ならば参考文献文 c は候補文献データ d を指し示す（すなわち x は正例）と判別し、さもなくば、c と d は一致しない (x は負例) と判別する。

このような同定手法で問題となってくるのが尺度につける重み ($\vec{w}$) と閾値 (b) の決め方である。調整すべき変数の数 ($n + 1$) が少ない場合においては、適切な重みを探すことは比較的易しい。しかし、尺度を増やすとその数だけ調整すべき変数が増えるため、手作業による重み付けは現実的ではない。そこでサポートベクターマシーン (SVM [7]) によって自動的に尺度の重みと閾値を決定する。

3.2.1 SVM の適用

教師つき機械学習アルゴリズムの一種である SVM は、マージン最大化基準を用いた 2 値分類器である。SVM は特徴量空間に写像された各事例を、超平面で正例、負例に分離する。この際、分離超平面に最も近い事例と分離超平面との距離（マージン）を最大にするような分離超平面を SVM は選択する。

また、事例を正例と負例に線形分離できない場合でもスラック変数を導入することで分離超平面を求めたり、過学習を抑えたりすることができる。スラック変数は分離できない事例と分離超平面の距離に比例する値で、SVM はこれらの事例のスラック変数の総和を小さくするように学習する。ただし、この総和を抑えることとマージンを最大にすることはトレードオフの関係にある。

今回、特徴量としては 4 節で述べる各尺度における参考文献文と文献データの類似度を使用した。具体的には、参考文献文と文献データのそれぞれを単語や文字組を基底とするベクトル空間に写像して各尺度の類似度を求める。今度は、各尺度における参考文献文と文献データの類似度を基底とするベクトル空間に写像し、その値を特徴量として SVM を使って分離平面を学習する。単語などを基底とするベクトル空間で直接分離せずに、一旦ベクトル同士の類似度を挟んでから分離する方式にしたのは、そのままでは訓練事例に現れない単語などを扱えないためである。SVM によって与えられる分離超

平面の方程式が、$\vec{w}^T\vec{x}+b=0$ であるとき、係数 $\vec{w}$,b がそのまま、判別関数 (式 (1)) における各尺度への重み $\vec{w}$ と閾値 b となる。

3.2.2 後処理

最後に、判別処理には後処理が必要である。というのも、候補事例の SVM による分類処理は、他の事例とは独立に行なわれる。ここで言う事例とは参考文献文と文献データを表すことに注意してもらいたい。したがって、一件の参考文献文に対して、複数候補があった場合、SVM はそれらをいずれも同時に正例と判定してしまう可能性がある。実際は、参考文献文はただ一つの文献データを指すので、同一の参考文献文を含む事例の中で、正例であるものは多くとも一つである。SVM が一件の参考文献文に対し複数の文献データを正解と判定してしまった場合、これらの中で各尺度における類似度の総和、$\vec{w}^T\vec{x}+b$ が最大のもののみを正解とし残りはシステム全体としては負例と判定する。

4 使用尺度

複数尺度を用いた同定では、3.1 節の絞り込み処理同様、ベクトル空間モデルを利用する。ただし、単語 1-gram 以外に複数の基底も用いる。加えて、文献データのフィールド情報も利用し、文献データ全体のベクトルの他に、個々のフィールドからなるベクトルと参考文献文ベクトル[1]の類似度も計算する。さらに、自分の類似度と他の候補の類似度とを比較する尺度も利用する。

以下利用した 1248 個の内訳を以下に述べる。

4.1 複数のベクトル空間

参考文献文と文献データの文字列を同一のベクトル空間に写像する。写像先のベクトル空間は一つではなく、用いる基底の違いによって 26 種類存在する。

単語 n-gram (9 種類) 各ベクトル空間の基底は単語（の組）である。ただし、単語を構成する文字種（大文字、数字）を利用して、特定の単語のみを基底に持つベクトル空間にも写像した。具体的には単語 1-gram から 3-gram、大文字のみの単語 1-gram から 3-gram、数字のみの単語 1-gram から 3-gram である。

文字 n-gram (11 種類) 各ベクトル空間の基底は固定長 n の文字列である。ただし、文字種（大文字、数字）を利用して、特定の文字のみを基底に持つベクトル空間にも写像した。具体的には、文字 2-gram から 6-gram、大文字 1-gram から 3-gram、数字 1-gram から 3-gram を使用した

単語情報と文字 n-gram の併用 (6 種類) 各ベクトル空間の基底は文字列であるが、与えられた文字列から単語の情報を利用した特定の文字を取り出し、その後、ベクトル空間に写像した。具体的には、単語の頭文字 1-gram から 3-gram、大文字と単語の頭文字 1-gram から 3-gram を使用した。

これらのベクトルの各要素の値は該当する素性が対象の参考文献文あるいは文献データに含まれるかに応じて 0 か 1 の二値で表した。また、単語の区切りには空白を用いたが、元からある空白に加えて、以下の方法でさらに空白を追加した。

- 記号を空白に変換する。
- 文字から数字の間、数字から文字の間、小文字から大文字の間に空白を加える。

これは、参考文献文と文献データの間の表記の揺れに対応するためである。また、参考文献文と文献データは大文字 n-gram の情報を抽出した後、全ての大文字を小文字に変換した。これも、参考文献文と文献データの間で大文字と小文字の使い方に一貫性があるとは限らないためである。

4.2 ベクトル間の類似度の尺度

以下の四つの尺度を用いる。参考文献文と文献データのベクトルをそれぞれ $\vec{c}$、$\vec{d}$ とする。

1. cosine
$$\frac{\vec{c}\cdot\vec{d}}{|\vec{c}||\vec{d}|}$$
2. 内積を参考文献文のベクトルの大きさの二乗で正規化したもの
$$\frac{\vec{c}\cdot\vec{d}}{|\vec{c}|^2}$$
3. 内積を文献データのベクトルの大きさの二乗で正規化したもの
$$\frac{\vec{c}\cdot\vec{d}}{|\vec{d}|^2}$$
4. 尺度 1. と尺度 2. のうち大きい方
$$\max(\frac{\vec{c}\cdot\vec{d}}{|\vec{c}|^2},\frac{\vec{c}\cdot\vec{d}}{|\vec{d}|^2})$$

[1]参考文献文をフィールドにあらかじめ分割しておくことも考えられるが今回は行なわなかった

尺度 2. と尺度 3. の違いを説明する。cosine (尺度 1.) は参考文献文と文献データの両方に現れる単語の数が多くなると値は大きくなり、一方のみに現れる単語が多くなると値は小さくなる尺度である。つまり、単語が過不足なく一致すると値が大きくなる。それに対して、内積を参考文献文ベクトルの大きさの二乗で正規化した値 (尺度 2.) は、両方に現れる単語の数が多くなると値は大きくなり、参考文献文のみに現れる単語の数が多くなると値が小さくなる点は同じであるが、文献データのみに現れる単語が多くなっても値の大きさには影響がないという点で異なる。参考文献文と参考文献文の指す文献データの単語が、必ずしも過不足なく対応とれるとは限らないことへの対応である。たとえば、「Annual Meeting of the Association for Computational Linguistics」と「Annual Meeting of the ACL」は、表記は異なるが同じ意味である。前者は 8 単語後者は 5 単語からなり、共通単語数は 4 である。したがって、尺度 1. の尺度では、$4/(\sqrt{8}\sqrt{5}) \simeq 0.63$ となるが、尺度 2. では、$4/5 = 0.8$ となる。文献データベクトルの大きさの二乗で正規化した尺度 3. を用いる理由は、今回用いたデータベースの不備のため、文献データより参考文献文の方が多くの情報を持つことがあるからである。なお、尺度 4. は尺度 2., 尺度 3. の大きな方を取る値である。これは 3.1 節で絞り込み処理に使用した尺度である。

4.3 フィールド情報の利用

参考文献文はフィールドには分かれていないが、文献データは著者、題目、掲載誌、掲載年、ページからなるフィールド情報を持つ。この情報を利用するため、文献データの各フィールドのベクトルと参考文献文全体とのベクトルの類似度を計算する。文献データの、著者、題目、掲載誌、ページ、掲載年の個々のフィールドの文字列これら五つのベクトルを足し合わせた文献データ全体からなるベクトルの合計 6 個のベクトルを用意し、それぞれ参考文献文のベクトルとの類似度を計算する。

4.4 最大値との差

参考文献文と文献データの各尺度における類似度の絶対的な大きさだけではなく、他のデータと比べた相対的な大きさも用いる。この相対的な大きさを表現する方法として、「最大値との差」を個々の類似度に対して定義する。これは、3.1 節で述べた特徴量 m_i を各尺度について適用するものである。参考文献文を固定した場合の類似度の最大値を自分が持てば値は 0 となるが、同じ尺度で自分より大きい類似度を持つ文献データが存在すると正の値を持つ。

4.5 まとめ

複数のベクトル空間を用いることで尺度の数を 26 倍、ベクトル間の類似度に複数の尺度を用いることで 4 倍、文献データの各フィールドのベクトルを用いることで 6 倍、各尺度について最大値との差も用いるとでさらに 2 倍して合計 1248 の尺度を用いる。

5 実験

複数尺度を用いた同定処理の有効性を検証するために実験を行なった。

5.1 実験条件

本実験は、著者の所属する研究室（奈良先端科学技術大学院大学 自然言語処理学講座）で管理されている文献データ[2]と OCR 読み取りされた参考文献文を対象とした。

文献データとして、自然言語処理、人工知能、情報検索、認知科学の関連文献の国際会議予稿集や論文誌から英語で書かれた、19786 件を用いた。文献データは、著者名、題目、掲載誌、ページ、掲載年、の各フィールドからなり、それぞれ人手にて入力されている。

OCR 読み取りされた参考文献文には、1997 年から 2001 年発行の自然言語処理関連の論文誌、会議予稿集から得られた 1707 の英語文献の参考文献一覧より、人手で各文に区切りを入れて得られた 30855 件から、無作為に 10000 件を選んだ。参考文献文は OCR の際、読み取り誤りが存在し、フィールド毎に分割されていない。参考文献文の指し示す文献が、文献データベース中に存在するものは 10000 件中, 3325 件であった。これは人手で調べた。

5 分割の交差検定により評価する。10000 件の参考文献文集合を無作為に 5 分割し、5 つの事例集合を生成した。5 つの事例集合の一つをテスト用のデータ、残りをトレーニングデータとした。

5.2 絞り込み

3.1 節の絞り込み処理は絞り込み後に残る負例の数と、誤って切り捨ててしまう正例の数を用いて評価した。

絞り込みの際に用いる各特徴量 (v, r, m)(3.1 節参照) の閾値はトレーニングデータ中の正解ペア (参考文献文と文献データとの同定ができているペア) 集合の特徴量から取得した。具体的には、正解ペア集合全てに対して各特徴量を求め、トレーニングデータ (8000×19786 事例) 各々の最低値をその特

[2] http://cl.aist-nara.ac.jp/mpd/

微量に関する閾値とした。

その結果、平均 665 件の正例を含んでいる各テスト事例 (参考文献文と文献データのペア)39572000 件 (2000×19786) を絞り込み処理を行なうことで、平均 2987.8 件まで減らすことができた。また、入力として与えられた正例、平均 665 件に対して、同定処理にまわされた正例は平均 664.2 件で各テストデータ毎の、誤って切り捨ててしまった正例の数は平均 0.8 件であった。

5.3 複数尺度を用いた同定手法の性能

絞り込み処理から複数尺度を用いた同定処理までの性能と、絞り込み処理から後処理までのシステム全体の性能についても評価した。

まず、実験の評価手法について述べる。

5.3.1 評価手法

実験は 5 分割交差検定で行なった。評価値には、

$$\text{適合率 (P)} = \frac{\text{TP}}{\text{TP} + \text{FP}}$$

$$\text{再現率 (R)} = \frac{\text{TP}}{\text{TP} + \text{FN}}$$

$$\text{F 値} = \frac{2\text{PR}}{\text{P} + \text{R}}$$

の 3 種類を用いた。ここで、TP、FP、FN、TN はそれぞれ以下の表で与えられる。

		正しい分類	
		正例	負例
SVM の判定	正	TP	FP
	負	FN	TN

適合率は負例をできるだけ選択しない評価値である。再現率は正例をもれなく選択する評価値で、参考文献の同定において、どちらの評価値がより重要かは明らかではない。評価値に F 値も用いた理由は、適合率と再現率に優先順位を付けられない以上、両方を同等に考慮する評価値が望ましいためである。

5.3.2 実験結果

表 1 に 5 分割交差検定の結果を示す。

表 1 の S「SVM」と書いてある行は複数尺度による同定手法自体の性能を表す。そのため絞り込み処理で切り捨ててしまった事例については評価しない。また、一つの参考文献文に対し複数の文献データを正例と判定した場合、正解の文献データがあれば正解である一つのペアを除いて不正解とした。なければ、全てのペアを不正解とした。

「絞り込み」と書いてある行は絞り込み処理と SVM による同定を続けて行なった場合の処理の性能を表す。絞り込み処理で切り捨ててしまった事例は、システムが負例として判定したものとする点が複数尺度自体に対する実験との違いである。

また、「システム」と書いてある行は絞り込み処理、SVM による同定、後処理の全てを行なった場合の性能を表す。つまりシステム全体の性能を表す。

各使用尺度に対する評価値は、SVM のマージンとスラック変数の総和の比を 0.01($=10^{-2}$),0.0312($=10^{-1.5}$),0.1($=10^{-1}$)...312($=10^{2.5}$),1000($=10^{3}$) と変化させ、最も F 値が高くなった時の値を用いた。

以下、具体的な実験方法について述べる。表 1 における使用尺度の「✓」はその列の尺度を使用したことを表す。

使用尺度の「最大値との差」は、4.4 節で説明した尺度を利用することを示す。この尺度を用いると使用尺度数は元の 2 倍になる。「文献データフィールド」は、4.3 節で説明したフィールド情報の利用を示す。使用尺度数は元の 6 倍になる。

「複数単語ベクトル」は 4.1 節で説明した単語 n-gram (9 種類) と単語情報と文字 n-gram の併用 (6 種類) の合計 15 個のベクトルで表すことを意味する。したがって、使用尺度数も元の 15 倍になる。

「複数文字ベクトル」は 4.1 節で説明した文字 n-gram (11 種類) ベクトルで表すことを意味する。したがって、使用尺度数も元の 11 倍になる。

使用尺度のベースラインには 4 個の尺度を用いた。ベクトルには単語 1-gram のみを用い、尺度には 4.2 節で説明した cosine などの 4 種類を用いた。

複数尺度を用いた同定手法自体の性能　使用尺度を増やすごとに F 値は大きくなっていることが分かる。全ての尺度を使用した時に、適合率、F 値が最も高くなった。再現率は、文献データのフィールド、複数の単語ベクトルと複数の文字ベクトルを用いたものと、全ての尺度を使用したものの二つで最も高い値を得た。

絞り込み処理から同定処理までの性能　次に絞り込み処理と複数尺度を用いた同定処理を連動して動かした時の性能であるが、これはほぼ複数尺度を用いた同定処理自体の性能と同じである。ただし、絞り込み処理でシステムが正例を負例と判定した事例が少数存在するため、複数尺度を用いた同定処理自体の性能より再現率が少し悪くなり、それにともない F 値も複数尺度を用いた同定手法自体のに比べ少し悪くなっている。

システム全体の性能　システム全体の性能は複数尺度を用いた同定手法自体の性能と同じように、全て

の尺度を使った時に、再現率が最も高くなった。しかし, 適合率は文献データのフィールド、複数の単語ベクトルと複数の文字ベクトルを用いたものが最も良い結果であった。F 値は、文献データのフィールド、複数の単語ベクトルと複数の文字ベクトルを特徴量として用いた時と、全ての尺度を使用した時、最も高く F 値で 0.992 となった。これは、十分に実用に耐える性能であるといえる。また、複数単語ベクトル、複数文字ベクトル、それに文献データのフィールド情報を特徴量で用いるだけでも全ての尺度を用いた場合と同等の精度が得られることが分かった。

6 考察

重みの分析 各尺度の類似度に重みを掛けてから足し合わせた総和が閾値を超えれば正例、超えなければ負例と判定した。各尺度についた重みの大きさや正負を比べることで、どの尺度が正例らしさや負例らしさに貢献しているかを知ることができる。そこで、最も 12 列目の実験条件の重みの傾向を調べた。

文字組と重みの中では、3-gram や 4-gram 程度の長さをみる尺度に大きな重みがついていた。反対に単語や単語の頭文字は 1-gram 等短い特徴量に重みがついていた。大文字の重みは正の重みも負の重みも小さな重みしかついていなかった。これは、文献データと参考文献文の両方に省略表記が含まれているとは限らないためと考えられる。そのかわり、「大文字と単語の頭文字」には重みがついた。数字や数字のみからなる単語は、掲載年、掲載誌の巻号、ページ、学会の開催回数などの一致を見る尺度として用いた。このような情報は、もし省略されなかった場合は正例の判定に有効だろうと予想したとおり、大きな重みがついていた。また、最大値との差についている重みは、負例らしさを表す尺度として負の重みがついていた。

使用する特徴量と性能の分析 表 1 を見ると、特徴量を追加するたびに判定の性能が上昇しているのが分かる。文献 [3] では、参考文献同士のクラスタリングタスクに置いては、単独で用いた場合単語特徴量の方が文字特徴量よりも有効であることが報告されている。本実験では文字列特徴量のみを用いた場合については評価を行なわなかったが、表 1 の列 (3) と列 (4)、あるいは、列 (11) と列 (12) を比較すると単語特徴量に対し文字列特徴量を加えてやることで性能が向上している（同じように列 (5), 列 (6) や列 (9), 列 (10) の実験結果も参照してもらいたい）。このことは文字列特徴量が単語に相補的な情報を含んでることを示している。

判定結果の分析 以下、判定結果の分析を行なう。なお、最も F 値が高かった実験条件で、正しい判定をできなかった事例の分析を行なう。

正例を負例と判定した誤りは 25 件存在した。その内訳を示す。なお、一つの事例に複数の誤りの原因がある場合は全てを加算した。

原因	回数
参考文献文の OCR 読み取り誤り	5
参考文献文と文献データの掲載誌の表記の相違、複数表示、省略	11
参考文献文の題目の省略, 欠落	6
参考文献文の間違い (ページ、題目、出典)	6
文献データの不備	3

上記の表を見ても分かるように、掲載誌に関する判定間違いが多いことが分かる。これは、他のフィールドと比べ、掲載誌フィールドが省略など表記の揺れが起こりやすいことに起因すると考えられる。判定間違いを効率的に減らすには、まず掲載誌フィールドが省略など表記の揺れに対応する必要がある。

また負例を正例と判定した誤りは合計 22 件存在した。この判定間違いをしたペアは共通した文字列が、掲載誌が異なる、題目が異なる（別にその題目の論文が存在する）、論文誌自体に対する参照などの理由で間違いであった。

7 まとめ

複数尺度の類似度を用いた参考文献の同定手法を提案した。この類似度は、各尺度の類似度に重みを掛け合わせてから足し合わせた総和で表す。また、各尺度につける重みは SVM で学習した。そして、この類似度を用いて、参考文献文と文献データのペアが正例か負例かを判定した結果、F 値 0.992 の性能が得られた。

この性能が得られた理由は二つある。一つは、参考文献文とそれが指す文献データの近さを適切に表す尺度を追加できたことがあげられる。先行研究では、使用尺度はたかだか数個にすぎず参考文献文と、文献データの情報を使い切っているとは言い難い。本手法では最大 1248 個もの特徴量を使用し、参考文献文と、文献データの情報を多角的に分析することに成功した。

もう一つは、各尺度につける最適な重みを教師つき機械学習で付与できたことである。使用尺度の数は最大 1248 個あり、これだけの数の尺度に対する重みを適切に人手で付与するのは難しい。

今後の課題を述べる。まずは、掲載誌が省略表記されたことによる誤りを取り除くことである。この対処法の一つは、参考文献文において掲載誌が占める割合を尺度に加える方法がある。ヒューリスティックな手法で参考文献文の各フィールドを同定してその割合を尺度に加えることもできるが、単純には、同定処理において参考文献文の文字数を特徴

表 1: 実験結果

		(1)	(2)	(3)	(4)	(5)	(6)	(7)	(8)	(9)	(10)	(11)	(12)
使	最大値との差							✓	✓	✓	✓	✓	✓
用	文献データフィールド		✓		✓	✓			✓			✓	✓
尺	複数単語ベクター			✓	✓	✓	✓			✓	✓	✓	✓
度	複数文字ベクター				✓		✓				✓		✓
尺度数		4	24	60	104	360	624	8	48	120	208	720	1248
適	SVM	0.926	0.981	0.953	0.973	0.981	0.991	0.958	0.983	0.968	0.980	0.984	**0.993**
合	絞り込み	0.926	0.981	0.953	0.973	0.981	0.991	0.958	0.983	0.968	0.980	0.984	**0.993**
率	システム	0.975	0.985	0.975	0.989	0.985	**0.995**	0.963	0.984	0.972	0.989	0.987	0.994
再	SVM	0.930	0.979	0.959	0.979	0.981	0.992	0.959	0.981	0.978	0.985	0.989	**0.993**
現	絞り込み	0.929	0.977	0.958	0.978	0.980	0.991	0.958	0.980	0.977	0.984	0.988	**0.992**
率	システム	0.928	0.977	0.956	0.977	0.979	0.990	0.958	0.979	0.977	0.983	0.988	**0.991**
F	SVM	0.928	0.980	0.956	0.976	0.981	0.992	0.958	0.982	0.973	0.982	0.986	**0.993**
	絞り込	0.927	0.979	0.956	0.975	0.980	0.991	0.958	0.981	0.972	0.982	0.986	**0.992**
値	システム	0.951	0.981	0.966	0.983	0.982	**0.992**	0.960	0.982	0.974	0.986	0.988	**0.992**

量に加えることも有効であると考えられる。掲載誌が欠落していなければ、文字数が少なるほど掲載誌の占める割合は高くなるからである。

また、絞り込みを行なう処理を高速化させる必要がある。現在、参考文献文数を M、データベース中の文献データ数を N とすると、計算量が $O(N \times M)$ となる。この計算量ではデータベース中の文献データが巨大な場合、計算するのに長時間かかってしまうことが予想される。そこで、絞り込み処理部分の計算時間を短縮する手法を考えたい。

謝辞 本研究は一部、21 世紀 COE プログラム「ユビキタス統合メディアコンピューティング」の支援を受けた.

参考文献

[1] S. R. Buss and P. N. Yianilos. A bipartite matching approach to approximate string comparison and search. Technical report, NEC Research Institute, 1995.

[2] S. Hitchcock, L. Carr, S. Harris, J. M. N. Hey, and W. Hall. Citation linking: Improving access to online journals. In R. B. Allen and E. Rasmussen eds., *Proceedings of the 2nd ACM International Conference on Digital Libraries*, pp. 115–122, Philadelphia,PA,USA, 1997.

[3] S. Lawrence, K. Bollacker, and C. L. Giles. Autonomous citation matching. In O. Etzioni ed., *Proceedings of the Third International Conference on Autonomous Agents*, New York, 1999. ACM Press.

[4] S. Lawrence, C. L. Giles, and K. Bollacker. Digital libraries and autonomous citation indexing. *IEEE Computer*, 32(6):67–71, 1999.

[5] A. McCallum, K. Nigam, J. Rennie, and K. Seymore. Building domain-specific search engines with machine learning techniques. In *AAAI Spring Symposium on Intelligent Agents in Cyberspace*, 1999.

[6] L. R. Rabiner. A tutorial on hidden Markov models and selected applications in speech recognition. *Proceedings of the IEEE*, 77(2):257–286, 1989.

[7] V. N. Vapnik. *The Nature of Statistical Learning Threory*. Springer, 1995.

[8] 難波, 奥村. 多言語論文データベースを用いたサーベイ論文検出－サーベイ論文自動作成の実現に向けて－. 言語処理学会第８回年次大会発表論文集, pp. 531–534, 2002.

[9] 堀部, 新保, 松本. ベクトル空間モデルを用いた参考文献の同定. 情報処理学会研究会報告 2002-NL-152, pp. 161–168, 2002.

[10] 三平, 山本. 引用文献の同定. コンピュータソフトウェア, Vol.14, No.1, pp. 35–39, 1997.