分散表現を利用したCRFによる参考文献書誌情報抽出

松岡　大樹† 　太田　学† 　高須　淳宏†† 　安達　淳††

† 岡山大学大学院自然科学研究科　〒 700–8530 岡山市北区津島中 3-1-1

†† 国立情報学研究所　〒 101–8430 東京都千代田区一ツ橋 2-1-2

E-mail: †{matsuoka, ohta}@de.cs.okayama-u.ac.jp, ††{takasu, adachi}@nii.ac.jp

あらまし　膨大な文書が格納されている電子図書館を運用するためには，書誌情報データベースの整備が必要である．特に，学術論文の参考文献欄には，タイトルや著者名などの有用な情報が集約されている．本研究のように，Conditional Random Field を用いて参考文献文字列から書誌情報を自動抽出する場合，利用する素性が抽出精度を決定する．これまでの研究により，書誌情報の高精度抽出には辞書が有効であることが分かっているが，辞書の作成コストの問題があった．そこで本研究では，辞書の代替として分散表現の利用を提案し，実験により参考文献書誌情報抽出におけるこの分散表現の適切性および書誌情報抽出精度を評価する．

キーワード　情報抽出，Conditional Random Field，参考文献文字列，辞書，word2vec，fastText

1. はじめに

多数の学術論文が蓄積されている電子図書館を快適に利用するためには，検索やソート，文書間リンク等の機能は必須といえる．しかし，そのための書誌情報を人手でデータベースに登録するコストは膨大なため，その作業を可能な限り自動化する文書解析技術が求められている．特に学術論文の参考文献欄には，関連する文献の情報が集約されており，そのタイトルや著者名などの書誌情報は有用である．

本研究では川上ら [1] と同様に，Conditional Random Field（CRF）[2] を用いて，参考文献文字列から書誌情報を自動抽出する．CRF を用いた参考文献書誌情報抽出においては，利用する素性が書誌情報の抽出精度を決定する．我々は，どのような素性が書誌情報の高精度抽出に有効であるのか検討し，辞書が有効であることを確認した [3]．しかし，著者名，論文誌名，会議名等の様々な辞書が存在し，それぞれの辞書のエントリを収集する必要があるため，その作成コストが問題となる．そこで本研究では，参考文献文字列を構成するトークンの分散表現が辞書の代替となるかを検討する．

本稿の構成は次の通りである．2 節で学術論文からの書誌情報抽出に関する研究を紹介し，続く 3 節で本研究で行う CRF による参考文献書誌情報の自動抽出について説明する．4 節で分散表現とそれを利用した代替辞書の作成手順について述べ，5 節で実験による評価を行う．最後に 6 節で本稿をまとめる．

2. 関連研究

多数の学術論文を格納する電子図書館において，書誌情報の管理は必須である．また学術論文からの書誌情報抽出では，ルールや機械学習がよく用いられる．ルールを用いて論文の参考文献文字列から書誌情報を抽出する場合，書式が異なる論文誌ごとに抽出ルールを設定する必要がある．これは，図 1 のように，著者名，タイトル，発行年などの書式が，論文誌ごとに異なるからである．

電子情報通信学会論文誌

J.L. Flanagan, J.D. Johnston, R. Zahn, and G.W. Elko, "Computer-steered microphone arrays for sound transduction in large rooms," J. Acoust. Soc. Am., vol.78, no.5, pp.1508-1518, Nov. 1985.

情報処理学会論文誌

Lipasti, M.H. and Shen, J.P.: Exceeding the Dataflow Limit via Value Prediction, Proc. 29th Annual International Symposium on Microarchitecture, pp.227-237 (1997).

図 1　学術論文誌による参考文献文字列の書式の違い

しかし，近年では学術論文誌が増え，論文誌ごとにルールを設定し，管理することが困難になりつつある．そのため，学習データを準備すれば，どの学術論文にも対応できる機械学習が注目されている．

その中でも CRF を利用した書誌情報抽出に関する研究として，Peng ら [4]，Councill ら [5]，そして Do ら [6]，Cuong ら [8] の研究がある．Peng ら [4] や Councill ら [5] は，HMM や CRF を用いて書誌情報を抽出した．Peng ら [4] はタイトルページと参考文献欄の単語に書誌要素ラベルを付与した．タイトルページからの書誌情報抽出では，英語論文 935 件を対象に，500 件を学習データ，435 件をテストデータとして実験を行った．著者名やタイトル，所属など 13 項目の書誌情報を抽出し，その F 値の平均は 0.939 であった．一方，参考文献欄からの書誌情報抽出においては，英語論文 500 件を対象に，350 件を学習データ，150 件をテストデータとして実験を行った．著者名や論文誌名，日付など 13 項目の書誌情報を抽出し，その F 値の平均は 0.915 であった．また，Councill ら [5] は，CRF に基づく書誌情報抽出ツールである ParsCit を開発し，参考文献文字列から書誌情報を抽出した．ParsCit では，空白文字をデリミタとして英文の参考文献文字列をトークン列に変換し，そのトークン列に書誌要素ラベルを付与する．彼らの実験は，Cora データセット [9] を対象に，著者名やタイトルなど 13 項目の書誌情報

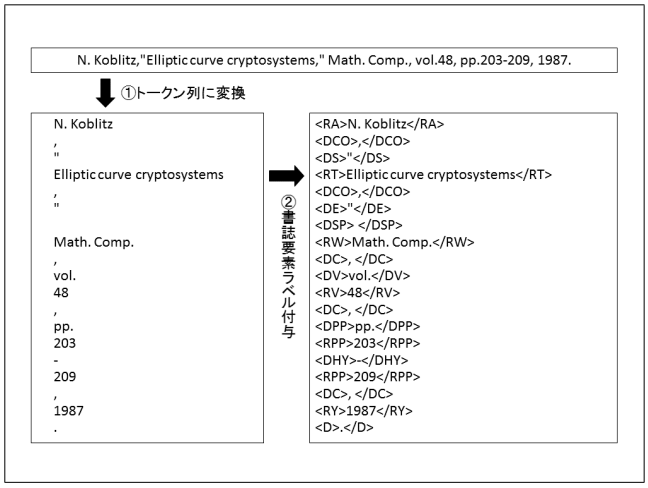

図 2　参考文献書誌情報抽出の例

を抽出し，その F 値の平均は 0.950 であった．

Do ら [6] は，学術論文の PDF から，著者名と所属の関係を抽出する情報抽出器である Enlil を開発した．Enlil は，まず CRF を利用して著者と所属を抽出し，その後 SVM [7] を使用してそれらを関連付ける．この抽出器では CRF による文字列からの情報抽出の際，トークンの長さや n-gram といった内容素性だけでなく，添え字や上付き文字などのレイアウト素性も利用する．Cuong ら [8] は linear-chain CRF（L-CRF）を改良した higher order semi-Markov CRF（HO-SCRF）を用いて，参考文献文字列解析，一般的な節のラベル付け，著者と所属の抽出の 3 つのタスクに取り組んだ．HO-SCRF は 2 つ以上の入力系列に対応して素性を作成し，書誌情報を抽出できる．実験は，L-CRF および，入力系列の数を変えた HO-SCRF を用いて行い，HO-SCRF はいずれのタスクにおいても L-CRF よりも良い結果を示した．

3. CRF による参考文献書誌情報抽出

3.1 参考文献書誌情報抽出

本研究では，学術論文の参考文献文字列から書誌情報を自動抽出する．具体的には図 2 のように参考文献文字列をまずトークン列に変換し，その後トークン列から著者名やタイトルといった主要な書誌情報を抽出する．参考文献文字列から抽出する書誌情報の一覧とそれに対応する書誌要素ラベルを表 1 にまとめる [1]．表 1 の Other は他のどの書誌要素にも分類されない書誌要素であり，具体的には所属機関などが含まれる．本研究では図 2 に示すように，トークン列の各トークンに対して RA や RT などの書誌要素ラベル，または DC などのデリミタラベルを付与する．なお，図 2 で D から始まるラベルはデリミタラベルを表し，DC(カンマ+空白) などが定義されている [1].

3.2 CRF

本研究の書誌情報抽出では，標準的な linear-chain CRF [2] の定義を用いて，参考文献文字列をトークン列に変換し，そのトークン列に書誌要素ラベルを付与する．また CRF では，入力系列 $\mathbf{x} = x_1, \ldots, x_n$ が与えられたとき，出力ラベル系列が $\mathbf{y} = y_1, \ldots, y_n$ となる条件付き確率を以下のように与える．

表 1　抽出する書誌情報 [1]

書誌要素	書誌要素ラベル
Author	RA
Editor	RE
Translator	RTR
Author Other	RAOT
Title	RT
Booktitle	RBT
Journal	RW
Conference	RC
Volume	RV
Number	RN
Page	RPP
Publisher	RP
Day	RD
Month	RM
Year	RY
Location	RL
URL	RURL
Other	ROT

$$P(\mathbf{y}|\mathbf{x}) = \frac{1}{Z_{\mathbf{x}}} \exp\left(\sum_{i=1}^{n} \sum_{k} \lambda_k f_k(y_{i-1}, y_i, \mathbf{x})\right) \tag{1}$$

ただし、$Z_{\mathbf{x}}$ は，全てのラベル系列を考慮したときに確率の和が 1 となるための正規化項で，

$$Z_{\mathbf{x}} = \sum_{\mathbf{y}' \in \mathbf{Y}(\mathbf{x})} \exp\left(\sum_{i=1}^{n} \sum_{k} \lambda_k f_k(y'_{i-1}, y'_i, \mathbf{x})\right) \tag{2}$$

である．ここで，$f_k(y_{i-1}, y_i, \mathbf{x})$ は $(i-1)$ 番目と i 番目の出力ラベルと入力系列 $\mathbf{x}$ に依存する任意の素性関数である．λ_k は素性関数 f_k の重みを表すパラメータで学習により定める．また，$Y(\mathbf{x})$ は入力系列 $\mathbf{x}$ に対する出力ラベル系列の集合である．そして，入力系列 $\mathbf{x}$ に対する最適な出力ラベル系列 $\mathbf{y}^*$ は次式で与えられる．

$$\mathbf{y}^* = \arg\max_{\mathbf{y} \in Y(\mathbf{x})} P(\mathbf{y}|\mathbf{x}) \tag{3}$$

本研究の書誌情報抽出では，ラベル付与の対象である入力 x_i は，参考文献文字列をトークン列に変換して得られるトークンであり，一方，ラベル y_i は，書誌要素またはデリミタのラベルである．本稿では，書誌要素ラベル付与の精度を評価するため，トークン列への変換は人手で行う．

3.3 素性テンプレート

本研究では工藤が作成した CRF++(注1) を利用して書誌情報を抽出する．CRF++で用いる素性テンプレートは川上らの素性テンプレート [1] を使用する．これを表 2 にまとめる．素性テンプレートについて説明する．この素性テンプレートは 47 種類の Unigram 素性と 1 種類の Bigram 素性の合計 48 種類の素性で構成されている．これらは全て言語的な素性で，レイアウトに関する素性はない．Unigram 素性には，トークンのトークン列における出現位置や文字数，トークンを構成する文字種とそ

（注1）：http://taku910.github.io/crfpp/

表 2　素性テンプレート [1]

種類	素性	数	内容
Unigram	<token_ab_pos(0)>	1	トークン列における絶対的な出現位置
	<token_re_pos(0)>	1	トークン列における相対的な出現位置
	<num_char(0)>	1	トークンの文字数
	<num_word(0)>	4	トークン内の単語数
	<num_period(0)>	4	トークン内のピリオド数
	<f_kanji(0)>	1	トークン内の漢字数の割合
	<f_hiragana(0)>	1	トークン内のひらがな数の割合
	<f_katakana(0)>	1	トークン内のカタカナ数の割合
	<f_alphabet(0)>	1	トークン内の全角アルファベット数の割合
	<f_digit(0)>	1	トークン内の全角数字数の割合
	<h_alphabet(0)>	1	トークン内の半角アルファベット数の割合
	<h_digit(0)>	1	トークン内の半角数字数の割合
	<h_symbol(0)>	1	トークン内の記号数の割合
	<first_1-4_string(0)>	4	トークンの先頭から 4 文字目までの文字列
	<last_1-4_string(0)>	4	トークンの末尾から 4 文字目までの文字列
	<token(0)>	1	トークン自身
	<last_char(*i*)>	1	トークンの最後の文字種
	<token_lc(*i*)>	1	トークンを小文字にした文字列
	<capital(*i*)>	1	トークン中の大文字の有無
	<digit(*i*)>	1	トークン中の数字の有無
	<symbol(*i*)>	2	トークン中の記号の有無
	<keyword(*i*)>	2	トークン中の特徴的な文字列の有無
	<dictionary(*i*)>	8	辞書素性
	<num_token(0)>	1	参考文献文字列のトークン数
	<editor(0)>	1	参考文献文字列中の Editor に関する記述の有無
	<URL(0)>	1	参考文献文字列中の URL に関する記述の有無
Bigram	< *y*(-1), *y*(0)>	1	ラベルの遷移

の割合，トークンの先頭・末尾から 4 文字目までの文字列，大文字などの特定の文字や特徴的な文字列，各種辞書のエントリの有無などがある．また，<dictionary(*i*)> における辞書としては，人名 (注2)，月名，地名 (注3)，出版社名 (注4)，論文誌名 (注5)，会議名 (注6) の辞書と，委員会名などをまとめた辞書の 7 種類の辞書を使用する．また，辞書素性には，どの辞書のエントリに一致したかを示す Dict という素性があり，この素性はヒットしたエントリを持つ辞書のビットを 1 とし，2 進数表現したものを 10 進数に直した素性である．

表 2 の各素性の括弧内の数字はトークンの相対位置を表し，0 が現在のトークンである．また $i \in \{-4,-3,-2,-1,0,1,2,3,4\}$ である．なお，表 2 で，“数”はその素性に関する要素数を表し，例えば，<first_1-4_string(0)> の場合，トークンの先頭の文字，先頭から 2 文字目までの文字，先頭から 3 文字目までの文字，先頭から 4 文字目までの文字という 4 つの要素をもつ．また，書誌要素ラベルの遷移を考慮するため Bigram 素性を用いる．この素性は付与される書誌要素ラベルの連接に関する情報を表し，これにより，例えば，著者名の後にタイトルがくるといった書誌要素の出現順に関する制約を考慮することができる．

4. 分散表現を利用した代替辞書

機械学習において文章中の単語を学習する手段として，単語ベクトルが利用されている．この単語ベクトルは，文章における単語の出現回数や単語の共起を利用して作成される．その中でも，ニューラルネットワーク（NN）を利用して，文章や単語の共起から単語の分散表現を獲得する手法が着目されている．本稿では，参考文献文字列を構成するトークンの分散表現を獲得し，それを代替辞書とする方法を提案する．

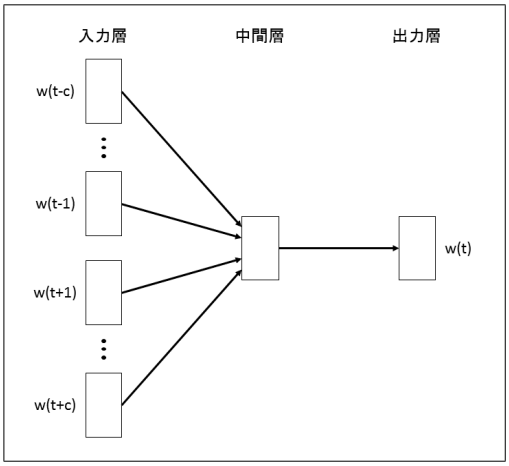

図 3　CBOW モデル

具体的には，学習データの参考文献文字列のトークンの分散表現を書誌要素ごとに集めたものを代替辞書とする．この代替辞書との照合は，トークン間の分散表現の類似度によって行う．本節では，まずこの分散表現について簡単に説明し，続いて提案する代替辞書について述べる．その後実験により，この辞書の分散表現の適切性を書誌情報抽出精度により評価する．

4.1 分散表現

単語の分散表現を獲得するための手段として，word2vec と fastText を利用する．

4.1.1 word2vec

word2vec は Mikolov ら [12,13] が提案したもので，単語を特徴ベクトルとして表す．word2vec を利用することで，単語間の共起や意味を含んだ比較的低次元のベクトルを獲得することができるため，自然言語処理の分野において広く用いられている．Mikolov らは word2vec を実現する NN として，CBOW モデルや Skip-gram モデルを提案した．

CBOW モデル　図 3 に CBOW モデルの NN の構造を示す．このモデルは，入力層，中間層，出力層の 3 層からなり，文章中の周囲の単語 $w(t-c),...,w(t-1),w(t+1),...,w(t+c)$ からその中心の単語 $w(t)$ を推測する NN である．ここで c は同じ文脈として考慮する前後の単語数を示す．

Skip-gram モデル　図 4 に Skip-gram モデルの NN 構造を示す．このモデルも CBOW モデル同様，入力層，中間層，出力層の 3 層からなる．しかし，文章中のある単語 $w(t)$ からその前後の単語 $w(t-c),...,w(t-1),w(t+1),...,w(t+c)$ を推測する NN である．つまり，CBOW モデルとは逆の問題を NN に学習させる．ここで c は同じ文脈として考慮する前後の単語数を示す．

4.1.2 FastText

fastText は Bojanowski ら [14,15] が提案したもので，word2vec 同様，単語を特徴ベクトルとして表すことができ，学習モデルとして CBOW モデルや Skip-gram モデルを採用している．fastText が word2vec と異なる点は，活用形や複合語などを考慮することができる点である．word2vec では例えば，“go”,

(注2)：http://www.census.gov/genealogy/names/など
(注3)：http://www.fallingrain.com/world/index.html など
(注4)：http://www.narosa.com/nbd/PublisherDistributed.asp など
(注5)：http://science.thomsonreuters.com など
(注6)：http://www.allconferences.com/など

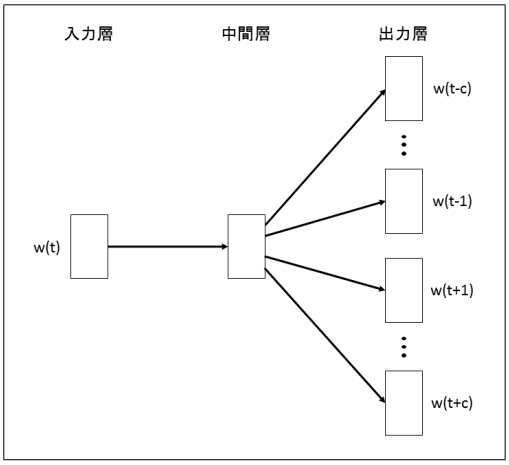

図 4　Skip-gram モデル

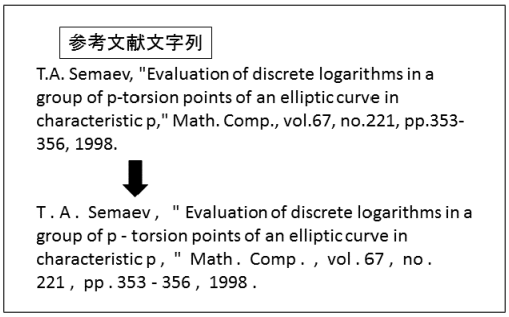

図 5　前　処　理

“goes”，“going” の間の関連性を考慮しないが，fastText ではこの関連性を考慮することができる．fastText には，subword という仕組みが用いられており，モデルを学習する際に，“goes” は “go”+“es”，“going” は “go”+“ing” といったように分割して学習する．これにより活用形を考慮する．

4.2　分散表現の獲得と代替辞書

4.2.1　分散表現の獲得

本稿では，word2vec および fastText を利用して，参考文献文字列に含まれるトークンの分散表現を獲得する．また実験では，この両者を比較する．word2vec および fastText のモデル作成には，電子情報通信学会英文論文誌（IEICE-E）と IEEE Trans. Computers（IEEE-CS）に含まれる参考文献文字列を学習データとして使用する．なお，カンマやピリオドなどのデリミタも文脈として考慮するため，前処理としてデリミタの前後に半角スペースを挿入してから学習する（図 5）．モデル作成後，トークンのベクトルは，そのトークンに含まれるワードの相加平均で算出する．ワードとは，トークンをスペースで区切った文字列である．例えば，図 5 の参考文献文字列にある第一著者の “T.A. Semaev” というトークンは，“T”，“.”，“A”，“.”，“Semaev” という 5 つのワードに分割される．よって，このトークンのベクトルは，この 5 つのベクトルの平均ベクトルとなる．

4.2.2　代 替 辞 書

4.2.1 項の方法で算出したトークンのベクトルを用いて代替辞書を作成する．実験では，表 1 に示した書誌情報を，[1] にならい，表 3 のように集約して正解判定を行うため，代替辞書もこれに対応するように作成した．よって，作成した代替辞書は，AUTHOR，TITLE，JOURNAL，VOLUME，PUBLISHER，DAY，MONTH，YEAR，OTHER の 9 種類である．まず CRF の学習に使用するラベル付き参考文献文字列から，トークンの分散表現を獲得する．そして獲得したトークンの分散表現を，付与されている書誌要素ラベルに基づき，各辞書に振り分ける．代替辞書の分散表現と参考文献文字列中のトークンが閾値以上類似していれば，その書誌要素の辞書に照合したとみなす．このときの類似度はコサイン類似度とし，このコサイン類似度が閾値を超えれば辞書に照合したと判定する．学習データから得た分散表現は，先に述べた 9 種類に分類して，これを 9 種類の代替辞書とみなす．このように作成した辞書を川上ら [1] の辞書と入れ替えて，書誌情報抽出精度を評価する．

表 3　書誌要素ラベルの大分類

書誌要素ラベル	分類名
RA, RE, RTR, RAOT	AUTHOR
RT, RBT	TITLE
RW, RC	JOURNAL
RV, RN, RPP	VOLUME
RP	PUBLISHER
RD	DAY
RM	MONTH
RY	YEAR
RL, RURL, ROT	OTHER

5.　評 価 実 験

5.1　実 験 概 要

4.2 節に示した分散表現の代替辞書を作成し，分散表現の適切性評価に基づいて，word2vec ならびに fastText の適当なパラメータを評価する．その後，そのパラメータで得た分散表現の代替辞書を用いて，CRF による書誌情報抽出精度を評価する．

実験データには，以下の英文参考文献文字列コーパスを利用する．

IEICE-E　2000 年の電子情報通信学会英文論文誌に含まれる参考文献文字列 4,497 件

IEEE-CS　1952 年から 2012 年までの IEEE Trans. Computers に含まれる参考文献文字列の引用回数上位 4,770 件

分散表現を獲得する際のモデルの作成には，IEICE-E および IEEE-CS の 2 論文誌を使用し，IEICE-E の論文誌の参考文献文字列からのみ分散表現を獲得する．そして，得られたトークンの分散表現の適切性を，word2vec と fastText のそれぞれについて，トークン間の書誌要素（表 1）またはデリミタラベルの一致率によって評価する．具体的には，IEICE-E に含まれる参考文献文字列のトークンから重複を削除し，その後，このユニークなトークンに対し，最も類似したトークンの書誌要素またはデリミタラベルを比較する．本研究では，このとき書誌要素またはデリミタラベルが一致すれば，その分散表現は適切であるとみなす．そして，ラベルの一致したトークン数を全トークン数で割ったものが一致率である．word2vec と fastText のそれぞれでモデルを作成する際，パラメータを変化させて，モデルごとの一致率を比較する．この一致率が高いパラメータほど適切

表 4　使用するパラメータ (word2vec)

パラメータ	説明	使用する値
size	出力するベクトルの次元数	50, 100（デフォルト）, 150
window	文脈として考慮する単語数	5（デフォルト）, 15, 25
iter	学習の反復回数	5（デフォルト）, 10, 15

表 5　使用するパラメータ (fastText)

パラメータ	説明	使用する値
dim	出力するベクトルの次元数	50, 100（デフォルト）, 150
lr	学習率	0.05（デフォルト）, 0.1, 0.15

表 6　トークンの一致率（word2vec）

size	一致率	window	一致率	iter	一致率
50	0.82128	5	0.82031	5	0.82031
100	0.82031	15	0.81893	10	0.84909
150	0.82134	25	0.79485	15	0.85591

表 7　トークンの一致率（fastText）

dim	一致率	lr	一致率
50	0.85804	0.05	0.85614
100	0.85614	0.1	0.84714
150	0.85723	0.15	0.84135

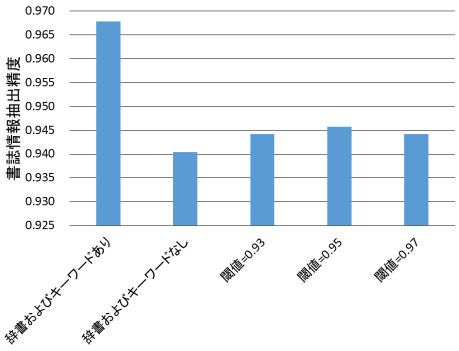

図 6　代替辞書を利用した書誌情報抽出精度（word2vec）

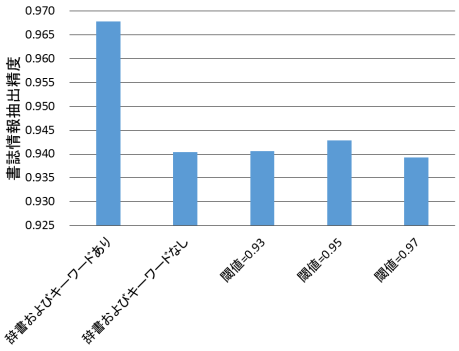

図 7　代替辞書を利用した書誌情報抽出精度（fastText）

な分散表現であるといえるため，その値を用いて代替辞書の分散表現を獲得する．

5.2　word2vec および fastText のパラメータ

word2vec および fastText のモデルを学習する際，それぞれのパラメータを変化させ，先に述べたトークンの一致率を利用して，パラメータの適当な値を探索する．変化させる word2vec のパラメータを表 4，fastText のパラメータを表 5 に示す．なお，これらの表で一つでのパラメータの値を探索するときは，他のパラメータの値は表中のデフォルトの値を用いる．また，表 4，表 5 に示したパラメータ以外のパラメータについては，min_count は 0 とし，その他はデフォルトの値を使用する．

表 4，表 5 に示すパラメータを用いて，IEICE-E の参考文献文字列に含まれるトークンの一致率を評価する．word2vec の結果を表 6，fastText の結果を表 7 に示す．

表 6 より，word2vec のパラメータは，size は 150，window は 5，iter は 15 のときが最も一致率が高く，また，表 7 より，fastText のパラメータは，dim は 50，lr は 0.05 のときが最も一致率が高かった．よって，これらのパラメータを使用して分散表現を獲得して，5.3 節の実験で使用する代替辞書とする．

5.3　代替辞書を用いた参考文献書誌情報抽出

5.2 節で説明した方法で作成した IEICE-E の分散表現の代替辞書を用いて，CRF による参考文献書誌情報抽出精度を算出する．書誌情報を抽出する実験データは，5.1 節に示した IEICE-E である．

書誌情報抽出の際の CRF++のパラメータはデフォルトの値を使用する．そして，辞書と照合したと判定する分散表現のコサイン類似度の閾値を 0.93，0.95，0.97 と変化させることにより，書誌情報抽出精度の変化を確認する．また，本実験においては，表 2 の <keyword(i)> のキーワード素性も辞書素性とみなし，辞書素性とキーワード素性を除いたときの書誌情報抽出精度と比較する．

word2vec で生成した代替辞書を用いた結果を図 6，fastText のそれを用いた結果を図 7 に示す．これらの図で “辞書およびキーワードあり” は，川上らの辞書 [1] を使用して書誌情報を抽出したときの結果である．また，“辞書およびキーワードなし” は，辞書素性およびキーワード素性を抜いて書誌情報を抽出したときの結果である．

図 6 より，word2vec を利用して作成した代替辞書を用いたときの書誌情報抽出精度は，コサイン類似度の閾値を 0.95 としたときが最も高い．このときの精度から，この代替辞書の性能は，川上らの辞書の性能の約 20%に相当することがわかる．また，図 7 より，fastText を利用して作成した代替辞書を用いたときの書誌情報抽出精度も，コサイン類似度の閾値を 0.95 としたときが書誌情報抽出精度が最も高い．このときの精度を見ると，fastText による代替辞書の性能は，川上らの辞書の性能の約 10%に相当している．よって，この実験からは，word2vec を使用して代替辞書を作成したほうが，fastText を使用して代替辞書を作成するよりも良い結果が得られた．しかし，川上らの辞書を使用した場合に比べて書誌情報抽出精度の上昇が小さいため，パラメータや閾値についてさらに検討が必要と考える．

作成した代替辞書の中で，どの辞書が有効であるか確かめるため，辞書を 1 種類ずつ除いて書誌情報抽出実験を行った．word2vec を用いて作成した代替辞書の比較実験の結果を図 8 に，fastText を用いて作成した代替辞書の比較実験の結果を図 9 に示す．図 8 より，VOLUME や MONTH の辞書を除いたとき

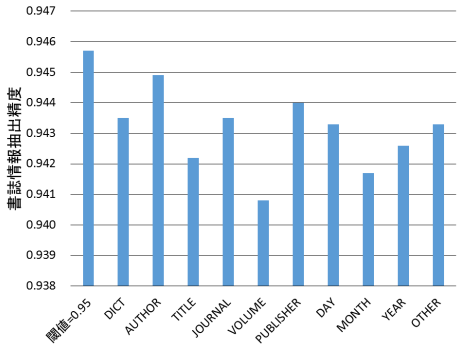

図 8　有効な代替辞書（word2vec）

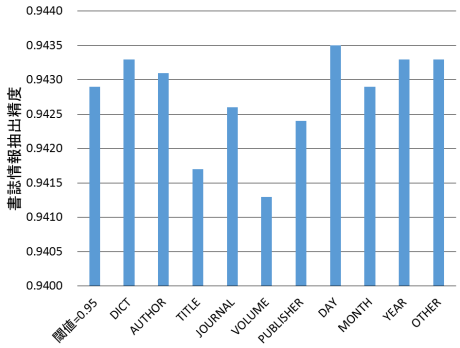

図 9　有効な代替辞書（fastText）

に抽出精度が大きく低下しているため，これらが有効な代替辞書であることがわかる．また，図 9 では，TITLE と VOLUME の辞書を除いたときに抽出精度が大きく低下しているため，これらが有効な代替辞書であることがわかる．

6. ま と め

本稿では，CRF による参考文献書誌情報抽出に利用するため，参考文献文字列中のトークンの分散表現の代替辞書を提案した．実験では，その代替辞書を利用して参考文献書誌情報抽出を行い，その抽出精度を評価した．本研究ではまた，word2vec および fastText により生成したトークンの分散表現の書誌要素またはデリミタラベルの一致率を用いて分散表現の適切性を判定した．この代替辞書を従来から使用している川上らの辞書と入れ替えて書誌情報抽出精度を評価したところ，word2vec による代替辞書では，川上らの辞書の約 20%，fastText による代替辞書では，川上らの辞書の約 10%に相当する性能を示した．提案した分散表現の代替辞書は，効果は確認できたものの，通常の辞書に比べそれは小さかった．そのため，分散表現を獲得するためのモデルのパラメータチューニングや，辞書との照合判定に用いる閾値について，今後さらに検討する必要があると考える．また，実用的な辞書の整備や日本語トークンの代替辞書の作成も今後の課題といえる．

謝　　辞

本研究の一部は，国立情報学研究所公募型共同研究の援助による．ここに記して深謝する．

文　　献

[1] 川上尚慶, 太田学, 高須淳宏, 安達淳, “少量学習データによる参考文献書誌情報抽出精度の向上”, 情報処理学会論文誌データベース, vol. 8, no. 2, pp. 18–29, 2015.
[2] J. Lafferty, A. McCallum and F. Pereira, “Conditional Random Fields : Probabilistic Models for Segmenting and Labeling Sequence Data”, In Proc. of 18th International Conference on Machine Learning, pp. 282–289, 2001.
[3] D. Matsuoka, M. Ohta, A. Takasu, J. Adachi, “Examination of Effictive Features for CRF-Based Bibliography Extraction from Reference Strings”, In Proc. of ICDIM2016, pp. 243–248, 2016.
[4] F. Peng, A. McCallum, “Accurate Information Extraction from Research Papers Using Conditional Random Fields”, HLT-NAACL 2004, pp. 329–336, 2004.
[5] I. G. Councill, C. L. Giles and M. Y. Kan, “ParsCit: An Open-Source CRF Reference String Parsing Package”, In Proc. of language resource and evaluation conference, 2008.
[6] H. H. N. Do, M. K. Chandrasekaran, P. S. Cho and M.Y. Kan, “Extracting and Matching Authors and Affiliations in Scholarly Documents”, In Proc. of JCDL2013, pp. 219–228, 2013.
[7] C. Cortes and V. Vapnik, “Support-Vector Networks”, Machine Learning, vol.20, no. 3, pp.273-297, 1995.
[8] N. V. Cuong, M. K. Chandrasekaran and M.Y. Kan, “Scholarly Document Information Extraction Using Extensible Features for Efficient Higher Order Semi-CRFs”, In Proc. of JCDL2015, pp. 61–64, 2015.
[9] A. McCallum, K. Nigam, J. Rennie and K. Seymore, “Automating the Construction of Internet Portals with Machine Learning”, Information Retrieval, vol. 3, no. 2, pp. 127-163, 2000.
[10] M. Ohta, R. Inoue, A. Takasu, “Empirical Evaluation of Active Sampling for CRF-Based Analysis of Pages”, In Proc. of IEEE IRI 2010, pp. 13–18, 2010.
[11] M. Ohta, R. Inoue, A. Takasu, “Empirical Evaluation of CRF-Based Bibliography Extraction from Research Papers”, IADIS International Journal on Computer Science and Information Systems, vol. 7, no. 2, pp. 18–31, 2012.
[12] T. Mikolov, I. Sutskever, K. Chen, G. Corrado and J. Dean, “Distributed Representations of Words and Phrases and their Compositionality”, Advances in Neural Information Processing Systems, pp. 3111–3119, 2013.
[13] T. Mikolov, K. Chen, G. Corrado and J. Dean, “Efficient Estimation of Word Representations in Vector Space”, arXiv preprint arXiv:1301.3781, pp. 1–12, 2013.
[14] P. Bojanowski, E. Grave, A. Joulin and T. Mikolov, “Enriching Word Vectors with Subword Information”, arXiv preprint arXiv:1607.04606, 2016.
[15] A. Joulin, E. Grave, P. Bojanowski and T. Mikolov, “Bag of Tricks for Efficient Text Classification”, arXiv preprint arXiv:1607.01759, 2016.