同義語辞書作成支援システム

寺田　昭[†]・吉田　稔[††]・中川　裕志[††]

同義語の同定は，情報検索，テキストマイニングなどのテキスト処理を行う上で必要な作業である．同義語辞書を作成することにより，テキスト処理の効率や精度の向上を期待できる．特定分野における文書には，専門の表現が多く用いられており，その中には，分野独特の同義語が多量に含まれている．例えば，日本語の航空分野では，漢字・ひらがなだけでなく，カタカナ，アルファベット，およびそれらの略語が同義語として用いられている．この分野の同義語は，汎用の辞書に登録されていないものが多く，既存の辞書を使用できないので，辞書を新たに作成する必要がある．また，辞書作成後も常に新しい語が発生するので，辞書の定期的な更新が必要となるが，それを人手で行うのは大変な作業である．
本論文では，同義語辞書作成を半自動化するシステムを提案する．システムは，クエリが与えられると意味的に同じ候補語を提示する．辞書作成者は，その中から同義語を選択して，辞書登録を行うことができる．候補語のクエリに対する類似度は，同義語の周辺に出現する語の頻度情報を文脈情報とし，その余弦から計算する．文脈情報のみでは十分な精度が得られない場合，既知の同義語を知識としてシステムに与えることにより，文脈語の正規化を行い，精度を向上できることを確認した．実験は，航空分野の日本語のレポートを対象とし，システムの評価には平均精度を用いて行い，満足できる結果が得られた．

キーワード：同義語，文脈情報，辞書作成，文脈語の正規化

A System for Constructing a Synonym Dictionary

Akira Terada[†], Minoru Yoshida[††] and Hiroshi Nakagawa[††]

To identify a synonym is a necessary procedure for text processing such as information retrieval and text mining. We can expect to improve the proficiency and performance in text processing by constructing a synonym dictionary. Same words might possibly be used as a different meaning if the target field differs, so a synonym dictionary has to be constructed for each field. In some fields in Japanese, such as in aviation, synonym nouns include kanji/hiragana, katakana, alphabet and their abbreviations. Many of these words are not registerd in a general dictionary. In addition, as new words always come to be used, the dictionary update is a big issue.
In this paper, we propose a system for constructing a synonym dictionary. The system will return synonym candidates on the descending order of similarity against a query. A synonym can be easily registered in a dictionary by looking the synonym candidates generated by the proposed system. We define a context information as

† （株）日本航空, Japan Airlines Co., Ltd.
†† 東京大学情報基盤センター, Information Technology Center, The University of Tokyo

words frequency appearing around a target word. Then a similarity is calculated by cosine measure using context information. We confirmed that the system performance was remarkably improved by providing the system with known synonym set to make context word nominalization, especially when the performance was low. We experimentally evaluated the system performance by aviation safety reports in Japanese and evaluated it by average precision, and got promising results.

Key Words: *Synonym, Context information, Constructing a dictionary, Context word nominalization*

1 はじめに

企業内には，計算機で処理できる形での文書が大量に蓄えられている．情報検索，テキストマイニング，情報抽出などのテキスト処理を計算機で行う場合，文書内には，同じ意味の語句（同義語）が多く含まれているので，その処理が必要となる．例えば，日本語の航空分野では，「鳥衝突」を含む文書を検索したい場合，「鳥衝突」とその同義語である「Bird Strike」が同定できなければ，検索語として「鳥衝突」を指定しただけでは，「Bird Strike」を含み「鳥衝突」を含まない文書は検索できない．したがって，同義語の同定を行わないと，処理能力が低下してしまう．

特定分野における文書には，専門の表現が多く用いられており，その表現は一般的な文書での表現とは異なっている場合が多い．その中には，分野独特の同義語が多量に含まれている．これらの多くは汎用の辞書に登録されていないので，汎用の辞書を使用することによる同義語の処理は難しい．したがって，その分野の同義語辞書を作成する必要がある．本論文では，このような特定分野における同義語辞書作成支援ツールについて述べる．

本論文では，特定分野のひとつとして航空分野を対象とするが，航空分野のマニュアル，補足情報，業務報告書等に使用される名詞に限っても，漢字・ひらがなだけでなく，カタカナ，アルファベットおよびそれらの略語が使用されている．例えば，飛行機のマニュアルの場合，「Flap」を日本語の「高揚力装置」と表現しないで「Flap」と表現し，用語の使用がマニュアルよりも自由なマニュアル以外の文書では，「Flap」や「フラップ」と表現している．また，略語も頻繁に使用され，「滑走路」を「RWY」，「R/W」と表現している．そして，これらの表現が混在している．その理由は，海外から輸入された語句は，漢字で表現するとイメージがつかみ難いものがあるためであり，そのような語句は，英語表現や英語のカタカナ表現が使用される．「Aileron」を「補助翼」というよりは，「Aileron」や「エルロン」と通常表現している．マニュアルの場合は，ある程度，使用語が統一されているが，マニュアル以外のテキストは，語句の使用がより自由で，同義語の種類・数も多くなっている．そして，分野の異なる人間や計算機にとって理解し難いものとなっている．

このようなテキストを計算機で処理する場合には，同義語辞書が必要であるが，これらの語句は，前述したように汎用の辞書に載っていない場合が多い．さらに，語句の使用は統制されているものではなく，また，常に新しい語が使用されるので，一度，分野の辞書を作成しても，それを定期的にメンテナンスする必要がある．これを人手だけで行うのは大変な作業である．

我々は，同義語の類似度をその周辺に出現する語句の文脈情報により計算することにより同義語辞書を半自動的に作成するツールを開発している (寺田，吉田，中川 2006, 2007)．本論文では，上記の支援ツールを基礎にした計算機支援による同義語辞書作成ツールを提案する．その動作・仕組みは以下の通りである．計算機は，与えられたクエリに対して，意味的に同じ語句（同義語）の候補を提示する．辞書作成者は，クエリをシステムに与えることにより，同義語の候補語をシステムから提示され，その中から同義語を選択して，辞書登録をすることができる．システムは，これまで蓄えられた大量のテキスト情報を参照し，与えられたクエリの文脈と類似する文脈を持つ語句を同義語候補語とする．文脈は，クエリ・同義語の候補語の周辺に出現する語句を使用している．既知の同義語が存在する場合には，これらの同義語を使用して文脈語を同定することにより，システムの精度向上を行った．提案手法は，語句を認識できればよいので，分野・言語を問わないものである．

実験は，日本語の航空分野のレポートを使用した．このコーパスには，上述したように多数の同義語が存在し，その多くは汎用の辞書に載っていないものである．

評価は，回答の中で正解が上位にある程，評価値が高くなる平均精度を用いて行い，他の手法と比較して満足できる結果が得られた．

論文構成は，第 2 節では関連研究について述べる．第 3 節では類似度と平均精度について述べるが，その中で文脈情報，類似度，平均精度の定義について説明する．第 4 節では提案方式の詳細と実験について述べる．コーパス，評価用辞書，特徴ベクトルの定義について説明し，文脈語の種類・頻度，window 幅による精度比較について述べる．第 5 節では，第 4 節の結果をもとにして，詳細な議論を行う．クエリ・同義語候補語の種類による精度の比較，大域的文脈情報との比較，文脈語の正規化，特異値分解，関連語について述べる．第 6 節では複合名詞の処理を述べる．複合名詞については，専門用語自動抽出システム (中川，森，湯本 2003) が抽出した複合名詞を使用することにより単名詞と同様の処理を行った．第 7 節では同義語辞書の作成について考察する．第 8 節では結論と今後の研究課題について述べる．

2　関連研究

同義語を自動的に計算する研究は，これまで数多く行われてきた．その種類としては，カタカナと英語の対応，英語とその略語の対応，日本語とその略語の対応などがある．略語処理では，略語の近傍に括弧書きで略語の定義がされている場合の研究がある (Schwartz and Hearst

2003), (Pustejovsky, Castao, Cochran, Kotecki, Morrell, and Rumshisky 2001). この手法は，略語の定義が略語の近傍でされているものについては有効であるが，文書の中で必ずしも略語の定義がされているとは限らない．本論文で扱う文書では略語の定義はされていないので，この手法は適用できない．カタカナとアルファベット（英語）の対応では，Knight らは，カタカナとアルファベット（英語）の対応を発音記号から対応付けしている (Knight and Graehl 1998). 阿玉らは，カタカナのローマ字表記とアルファベットとの対応付けをしている (阿玉，橋本，徳永，田中 2004). Terada らは，英語における原型語とその略語の対応を両者に含まれる文字及びその順序などの情報を使用することで同定している (Terada, Tokunaga, and Tanaka 2004). この研究も本論文と同じく，航空分野という特定分野を対象としているが，対象とする言語が英語であり，略語をその原型語に復元するタスクを目的としている．

同義語の類似度の計算は，文脈情報から余弦を用いて計算するものが多い．文脈情報として，語句の前後の局所的なものを用いるもの (Terada et al. 2004), 文書全体から抽出して用いるものがある (酒井，増山 2005). Ohtake らは，カタカナの変形を探すのに，エディット距離で候補を絞った後に，文脈情報を用いているが，その際，カタカナが用いられている構文を解析して，動詞，名詞，助詞を使用している (Ohtake and Sekiguchi 2004). Masuyama らは，カタカナ処理で WEB データから英語に対応するカタカナのエディット情報を取得している (Masuyama and Nakagawa 2005). 文脈情報を用いる場合には，全ての種類の語句を用いるのでなく，内容語を用いるものが多い．

計算量の削減及び精度の向上のために，文脈情報だけではなく，文字情報を用いて，対応関係を絞り込む，または，決定する研究が多い．

本論文では，日本語を対象とし，漢字，ひらがな，カタカナ，アルファベット，およびそれらの略語の類似度を同時に計算するために，文字情報による絞り込みは行わず，文脈情報のみでどの程度の精度が得られるかを実験した．Terada らは，英語を対象として，略語とその原型語の対応を文脈情報および文字情報を使用して行っているが (Terada et al. 2004), 略語とその原型語のみならず，その他の同義語においても文脈情報を使用することにより，クエリに対する同義語が得られると考えた．したがって，提案手法は，Terada らの手法を応用し，言語を日本語に適応し，対象を略語から同義語に拡張し，文脈情報の使用に工夫を加えたものである．また，Terada らは，略語復元の精度を向上させるために，略語の多いコーパスと略語の少ないコーパスを使用しているが，提案手法では，同義語が同一のコーパスに含まれている場合は，コーパスは 1 つでよいと考え，1 種類のコーパスのみを使用した．文脈情報のみを使用しているが，同義語の日本語の文字種（漢字，ひらがな，カタカナ，アルファベット）について，種類の組み合わせにより精度が異なるかを調べ，今後の精緻なシステム構築の参考となるようにした．さらに，文脈情報のみでは，十分な精度が得られない場合があるので，既知の同義語を知識として使用することにより，精度の向上を図った．

3　類似度と平均精度

システムは，クエリに対して同義語候補語を順位付けして出力する．そのためには，クエリに対する同義語候補語の類似度を計算できなければならない．本節では，同義語候補語，文脈情報を定義し，提案手法での類似度について説明する．本節では，単名詞の処理について述べ，複合名詞の処理については，第 6 節で述べる．

3.1　同義語候補語

単名詞の同義語候補語は，テキストを形態素解析し，形態素解析器が出力した名詞である，漢字・ひらがな，カタカナ，アルファベットとした．形態素解析器は茶筅[1] を使用し，その中で出現頻度が 100 以上のものを使用した．

3.2　文脈情報

「同義語は，同じような文脈で使用される」という仮定から，語句の類似度を文脈の類似性から計算できると考えた．これは，人間が語の意味を理解するのにその語句が出現する前後の文脈から類推しているというアイデアからである．文脈は，同義語の近傍の語句（局所的文脈）とした．人間は，前後の語句の中で，場面に応じて文脈語を選別をしていると考えられるが，計算機で実現するのは不可能であるので，場面に応じた選別については，この研究では考慮しないことにした．

クエリを q とし，その前後の語句の並びを，$x_\alpha \ldots x_2\ x_1\ q\ y_1\ y_2 \ldots y_\beta$ とする．ここで，前後の語句は，形態素解析器が出力した単語とする．対象とするクエリの文脈語をクエリの前で $x_\alpha \ldots x_1$，クエリの後ろで $y_1 \ldots y_\beta$ とすると，window 幅は α, β であり，これ以降 window $[\alpha, \beta]$ と表現することとする．同義語候補語の window 幅についても，同様とする．window 幅は，クエリ，同義語候補語を含む 1 文の範囲内だけを考慮した．どのような文脈語を選択するかについては，第 4.3 節で述べる．

3.3　類似度

クエリ (query) の文脈情報を $\boldsymbol{c_q}$，同義語候補語 (synonym) の文脈情報を $\boldsymbol{c_s}$ とする．$\boldsymbol{c_q}$ と $\boldsymbol{c_s}$ をベクトル空間モデルで表し，その類似度をベクトルの余弦で表すと，クエリと同義語候補語の類似度 (sim) は，次式で計算される．

$$sim(query, synonym) = \frac{\mathbf{c_q} \cdot \mathbf{c_s}}{|\mathbf{c_q}| \cdot |\mathbf{c_s}|} \tag{1}$$

[1] http://chasen.naist.jp/hiki/ChaSen/

3.4 平均精度

情報検索の性能評価として精度と再現率がよく用いられるが，これらは，与えられたクエリに対する検索結果全体に対する性能を表すものである．同義語の検索結果から辞書作成者が辞書登録することを考えると，検索結果の順位における精度が重要である．つまり，上位の検索結果ほど評価値は高い必要がある．このような評価尺度を表すものとして平均精度 (average precision) を用いた．N 個のクエリの評価をする場合，i 番目のクエリに対する平均精度は次式で表される:

$$AveragePrecision[i] = \frac{1}{R[i]} \sum_{j=1}^{N_s[i]} (rel[j] \cdot \sum_{k=1}^{j} rel[k]/j) \tag{2}$$

ここで，

$N_s[i]$: i 番目のクエリの同義語の候補数.

$R[i]$: i 番目のクエリの同義語数.

$rel[k]$: システムが順序付けした回答の中で，k 番目の回答が正解であれば 1，そうでなければ 0.

i 番目のクエリに対する平均精度は，検索結果の各順位での精度 $\sum_{k=1}^{j} rel[k]/j$ の同義語 i 番目全体に対する和を同義語数 $R[i]$ で割ったものである．

N 個のクエリ全体の平均精度は，次式のように個々のクエリに対する平均精度の平均として定義する：

$$AveragePrecision = \frac{1}{N} \sum_{i=1}^{N} AveragePrecision[i] \tag{3}$$

4 提案方式の詳細と実験

第 4.1 節では，実験に使用したコーパスの説明をする．第 4.2 節では，評価用に人手で作成した辞書について述べ，第 4.3 節では，提案手法で用いる特徴ベクトルについて述べる．第 4.4 節では，window 幅等による比較についての実験結果を示す．

4.1 コーパス

コーパスとして，日本語の航空分野のレポートを使用した．個人情報保護の観点から，事前に名前等の個人情報は削除し，個人を特定できないような処理を行った．レポートの内容には，出発地・到着地などの定型情報とテキストで自由に記述された表題，本文が含まれているが，本文を対象とした．1992 年から 2003 年までのレポートを使用した結果，6,427 件のレポートが対象となり，そのサイズは，約 6.9 M バイトであった．

同義語候補語は，第 3.1 節で述べたように名詞を対象とし，その中には，漢字・ひらがな，カ

タカナ，アルファベット，およびそれらの略語があるが，その頻度が 100 以上のものを対象とした．その結果，同義語候補語の数は，1,343 になった．同義語抽出のタスクは，クエリに対する同義語をこれらの同義語候補語の中から選択するものである．

4.2 評価用辞書

今回の実験評価のために，4.1 節と同じ条件で出現頻度が 100 以上の候補語の中から，人手で選んだ 406 個の単語に対する同義語を求めることにより同義語辞書を作成した．単語には，同義な語句が複数存在する場合があるので，406 個のクエリに対する同義語数は 777 になり，平均同義語数は 1.91 であった．同義語の中には，「Service」，「SVC」，「サービス」のようにアルファベット（英語）とその略語およびそのカタカナ表現のほか，「Traffic」，「相手機」のようにドメイン特有のものも含まれる．

4.3 特徴ベクトルの定義

文脈情報を特徴ベクトルとして表すが，類似度計算に使用する特徴ベクトルの定義には，様々な方法がある．本節では，特徴ベクトルの定義が精度にどのような影響を及ぼすかを調査した．クエリと同義語候補語の文脈語としてそれぞれの前後に出現する語句を用いるが，本節では，名詞（漢字・ひらがな，カタカナ，アルファベット），動詞，形容詞という内容語を使用した．クエリ・同義語候補語の文脈情報は，コーパス全体の中でクエリ・同義語候補語の window 内に出現する文脈語を取得し，その頻度ベクトルとした．

類似度は，3.3 節で述べたように余弦で計算する．クエリの文脈ベクトルを $\boldsymbol{c_q} = (q_1, \ldots, q_{N_c})$，同義語候補語[2] の文脈ベクトルを $\boldsymbol{c_s} = (s_1, \ldots, s_{N_c})$ とすると，類似度 (sim) は次式で表される（N_c: 文脈語の異なり数）：

$$sim(query, synonym) = \frac{\mathbf{c_q} \cdot \mathbf{c_s}}{|\mathbf{c_q}| \cdot |\mathbf{c_s}|} = \frac{\sum_{i=1}^{N_c} q_i s_i}{\sqrt{\sum_{i=1}^{N_c} q_i^2 \sum_{i=1}^{N_c} s_i^2}} \tag{4}$$

ここで，文脈ベクトルの各要素 (q_i 又は s_i) は，文脈語の頻度を対数で補正したものを表す．表 1 は，頻度の対数による補正の有無の比較を示すが，対数による補正が精度に与える影響が

表 1　文脈語の頻度の対数による補正の比較

	window [2,2]	window [3,3]
平均精度（%）（対数による補正なし）	27.3	28.3
平均精度（%）　（対数による補正あり）	43.1	39.2

[2] 今後，同義語候補語を候補語と呼ぶこととする．

表 2 文脈語の種類による比較 (window [2,2])

文脈語の種類	平均精度 (%)
名詞 (漢字・ひらがな, カタカナ, アルファベット)	40.5
名詞 (漢字・ひらがな, カタカナ, アルファベット), 動詞	42.3
名詞 (漢字・ひらがな, カタカナ, アルファベット), 動詞, 形容詞	43.1
名詞 (漢字・ひらがな, カタカナ, アルファベット), 動詞, 形容詞, 助詞	27.3

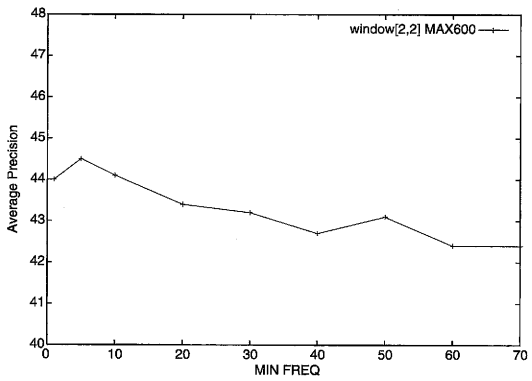

図 1 文脈語の最小頻度による平均精度への影響

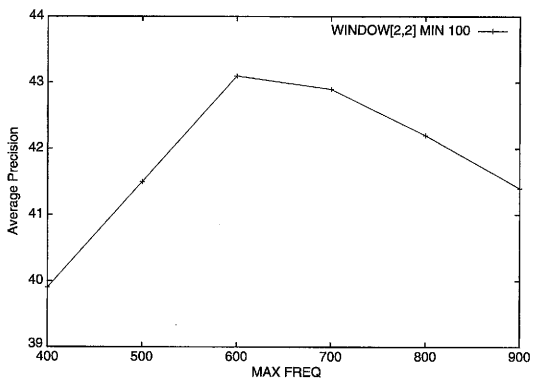

図 2 文脈語の最大頻度による平均精度への影響

大きいことが分かる.

文脈語として, 名詞 (漢字・ひらがな, カタカナ, アルファベット), 動詞, 形容詞を選択した場合とそれ以外の文脈語を選択した場合の比較を表 2 に示す.

文脈語の頻度については, 高頻度の文脈語は, 一般的であり同義語の判別に役立たず, 一方, 低頻度の文脈語は, 特殊すぎてノイズとなることが考えられる. したがって, 中程度の頻度の文脈語を採用するのがよいと考えられるので, 最小頻度 50, 最大頻度 600 を使用するものとする (図 1, 2 参照).

4.4 window 幅による比較

window 幅をどのように設定すれば, 平均精度が最適になるかを調査した. window 幅を大きくすれば, 候補語に対する文脈語を多く得られる反面, 候補語から遠い文脈語は, 候補語と関連性が薄くなり, ノイズとして悪影響を及ぼすので同義語の判別能力が弱くなり, また逆に, window 幅を小さくすれば, 候補語に対して得られる文脈語が少なくなり, 判別に使用する情報が少なくなると考えられる.

window 幅を同義語候補語の前 (FWD) に 0〜4 語, 後 (AFT) に 0〜4 語, 変化させて実験した結果を表 3 に示す. 平均精度は, window [2,2] が 43.1% で最も高かった.

同義語候補語の前後の window の比較では, window [2,0] では 35.5%, window [0,2] では 37.8%

表 3　window 幅による平均精度（%）の比較

FWD \AFT	0	1	2	3	4
0	—	21.4	37.8	37.1	33.0
1	25.6	31.5	40.7	38.7	35.3
2	35.5	36.9	43.1	40.1	36.8
3	34.8	37.8	40.9	39.2	36.1
4	32.2	34.6	37.4	36.3	33.3

であった．例えば，「Boarding」というクエリに対する正解は「搭乗」であるが，window [2,0] では「搭乗」が 1 位になるが，window [0,2] では 8 位になる．理由としては，window [2,0] では「Boarding」と「搭乗」の前に共通の語である「お客様」が多く出現するが，window [0,2] では「Boarding」と「搭乗」の後に共通の文字列（例：「を開始」など）の出現が少ないためであると考えられる．つまり window [2,2] では，window [2,0] の影響を受けて 1 位になっているといえる．「CAT[3]」というクエリに対する正解は「TURB」と「揺れ」であるが，window [2,0] では「TURB」が 2 位，「揺れ」が 9 位になる．共通に出現する代表的な言葉は「突然の」であるが，その数がそれ程多くないためだと考えられる．window [0,2] では「TURB」が 1 位，「揺れ」が 2 位になる．その理由として，後に「に遭遇」という表現が多く出現しているからだと考えられる．window [2,2] では，window [0,2] の影響を受けて「TURB」が 1 位，「揺れ」が 2 位になっている．

各同義語によりバラツキはあるものの，全体を通して，window [2,2] では，同義語の前 2 語の window [2,0] と同義語の後 2 語の window [0,2] が補完しあって，よい結果になっているものと考えられる．クエリ・候補語の window 幅内の前と後とでどちらが精度に貢献しているかについては，顕著な差は認められなかった．

5　議論

第 4 節での実験結果をもとにして，以下のような考察を行った．第 5.1 節ではクエリ・候補語の種類による精度の違いを調査した．第 5.2 節では，文脈情報の正規化による精度変化について述べる．第 5.3 節では，本手法がクエリ・候補語の近傍の文脈情報を使用しているのに対して，文書からの大域的情報を用いる手法との精度の比較を行った．第 5.4 節では関連語の検索について述べる．

[3] CAT は，Clear Air Turbulence を表す．

5.1 クエリ・候補語の種類による精度の違い

本節では，クエリと候補語の種類による精度の違いについて調べる．同義語の種類として，「Dispatch」と「DISP」のようなアルファベット同士，「ベルト」と「Belt」のようなカタカナとアルファベット，「座席」と「席」のような漢字同士，「Check」と「検査」のようなそれ以外のものに分類して，表 4 のような基準で平均精度を調べた．一般に，候補語の頻度が高いほど文脈情報が豊富となり，平均精度も高くなる傾向にあるため，候補語の頻度に対する閾値を増加させることで平均精度を上げることができる．このようにして平均精度を上げることで 50% を超えることができた場合，基準 3 に該当する．また，10% 以上 50% 未満のときを基準 4，10% 未満のときを基準 5 として分類した．また，閾値の頻度 100 未満でも 50% を超えることができた場合を基準 2，頻度 50 未満でも 50% を超えることが出来た場合を基準 1 とした．

表 5 にその結果を示すが，横軸の基準の数字は，各分類毎の基準 1〜5 での比率を示す．各分類の括弧の中の数字は，各分類の全体での比率を示す．アルファベット同士は，基準 1 と基準 2 の合計で 81% 以上の平均精度が得られた．カタカナとアルファベットでは，基準 1 から基準 3 までの合計でも平均精度は 15% 程度であった．この理由としては，カタカナとアルファベットでは，ごく近傍に出現する語の種類（カタカナではカタカナが多く，アルファベットではアルファベットが多い）が異なるためである．漢字同士の場合には，基準 1 から基準 3 までの合計で平均精度は約 63% 得られた．それ以外の場合は，全体の 76% を占めるが，基準 1 から基準 3 までの合計で平均精度は約 27% であった．

この結果から，カタカナとアルファベット及びそれ以外の分類のものの精度が低いことが分かった．それに対して，アルファベット同士，漢字同士の同義語の場合には，高い確度でユーザに同義語を提示できる．

表 4 同義語候補語の頻度による精度の高低の分類基準

基準 1	頻度 50 未満で，平均精度が 50% 以上
基準 2	頻度 100 未満で，平均精度が 50% 以上
基準 3	頻度を大きくすると，平均精度が 50% 以上
基準 4	頻度を大きくすると，平均精度が 10% 以上
基準 5	頻度を大きくしても，平均精度が 10% 未満

表 5 同義語候補語の種類による平均精度（%）の比較

分類 \ 基準	基準 1	基準 2	基準 3	基準 4	基準 5
アルファベット同士（13.4）	69.2	12.5	6.7	8.7	2.9
カタカナ　アルファベット（8.5）	4.5	3.0	7.6	18.2	66.7
漢字同士（2.1）	18.8	12.5	31.3	12.5	25.0
それ以外（76）	10.7	3.9	12.7	22.8	49.9

5.2 文脈語の正規化

第 5.1 節でカタカナとアルファベット及びそれ以外の分類のものの精度が低いことが分かったが，その解決法を考える．その方法として，文脈語に出現する同義語を同定することを考え，それによる精度変化を調べた．同義語同士の周辺に出現する文脈語を観察すると，文脈語の中にも同義語[4] が多く存在する．例えば，「Cargo」と「貨物」という同義語には，「Cargo Loading」と「貨物搭載」というように「Loading」と「搭載」という文脈同義語が出現する．しかしながら，「Cargo 搭載」，「貨物 Loading」という使用は，ほとんどされないので，「Cargo」と「貨物」の文脈語の中で「Loading」と「搭載」の分布は偏っている．したがって，特徴ベクトルにおいて別の要素である「Loading」と「搭載」は，「Cargo」と「貨物」の類似度の向上にあまり寄与しない．そこで，これらの文脈同義語を正規化[5] することにより平均精度の向上が期待できる．

実験として，筆者の 1 人が選択した 25 対の同義語（表 6 参照）について，41 個の文脈同義語（表 7 参照）の正規化を行い，個々の同義語の精度変化および評価辞書全体への影響を調査した．25 対の同義語の平均精度は，9.6% で，評価辞書全体の精度 43.1% と比較して難易度の高いものである．文脈同義語は，各同義語について特徴的なものを，筆者の 1 人が，PortableKiwi (藤本，吉田，中川 2005) を使用して 1〜4 個選択した．PortableKiwi は，対象としているコーパスに対して，ある言語表現を入力すると，その前後に現れる適当な長さの文字列 (Tanaka-Ishii and Nakagawa 2005) のうち，頻度の高いものから順に表示する用例検索システムである．

表 6 文脈語の正規化に使用した同義語対（括弧内の数字は，正規化に使用された文脈同義語の表 7 の番号を示す．最初の括弧は文脈語の頻度制限をしたもの，2 番目の括弧は頻度制限をしないもの．）

1. Belt ＝ベルト（18, 34）（18, 34）	2. Bird ＝鳥（15）（15）
3. Bus ＝バス（ ）（3）	4. ケース＝ Case（ ）（19）
5. Door ＝ドア（ ）（19）	6. Final ＝最終（27）（1, 2, 27）
7. Fuel ＝燃料（20, 35, 40）（17, 20, 35, 40）	8. LDG ＝着陸（6）（6）
9. Loading ＝搭載（7, 33）（1, 7, 33）	10. Plan ＝計画（27）（4, 9, 27, 28）
11. SVC ＝サービス（16）（16, 23, 26, 36）	12. Sign ＝サイン（12）（12）
13. Turn ＝旋回（5）（5）	14. Type ＝型（ ）（21）
15. Water ＝水（ ）（17）	16. キャンセル＝ Cancel（ ）（4, 9, 22）
17. センター＝ Center（13）（13）	18. ブリーフィング＝ BRFG（10, 41）（10, 37, 41）
19. Handling ＝ハンドリング（8, 39）（8, 39）	20. Cargo ＝貨物（11, 27）（11, 27）
21. Flight ＝フライト（38）（38）	22. PROC ＝手順（ ）（30）
23. Visual ＝目視（ ）（24, 29, 31）	24. グループ＝ GRP（ ）（23, 25, 36）
25. 出発＝ DEP（ ）（32）	

[4] 今後，文脈同義語と呼ぶこととする．
[5] ここで正規化とは，文脈語を既知の同義語に置換することをいう．例では，「搭載」を「Loading」に置換することである．

表 7 正規化に使用した文脈同義語（矢印（⇒）は，⇒の左の語句から右の語句に置換したことを示す.）

1. 燃料⇒ Fuel	2. 進入⇒ APP	3. ゲート⇒ Gate
4. Flight ⇒ FLT	5. 右⇒ Right	6. 重量⇒ WT
7. Cargo ⇒ CGO	8. Ground ⇒ GND	9. フライト⇒ FLT
10. Dispatch ⇒ DISP	11. Loading ⇒ Load	12. ベルト⇒ Belt
13. オペレーション⇒ Operation	14. ブリーフィング⇒ BRFG	15. 衝突⇒ Strike
16. 食事⇒ Meal	17. 漏れ⇒ Leak	18. シート⇒ Seat
19. クローズ⇒ Close	20. 補給⇒ Load	21. 改良⇒ Improved
22. 便⇒ FLT	23. 旅客⇒ PAX	24. 確認⇒ INSP
25. お客様⇒ PAX	26. 機内⇒ Cabin	27. 搭載⇒ Load
28. 飛行⇒ FLT	29. 点検⇒ INSP	30. 通常⇒ Normal
31. 検査⇒ INSP	32. 遅延⇒ Delay	33. 貨物⇒ CGO
34. 座席⇒ Seat	35. 残存⇒ Remain	36. Passenger ⇒ PAX
37. キャビン⇒ Cabin	38. プラン⇒ Plan	39. グランド⇒ GND
40. Remaining ⇒ Remain	41. ディスパッチ⇒ DISP	

最初に文脈語の頻度をこれまで通り 50 から 600 のものに制限すると，表 7 の 41 個の文脈同義語の内，使用されたものは 21 個で，表 6 の 25 対の同義語の中で文脈同義語が使用されたものは，15 対であった．ただし，15 対の同義語は，想定していた文脈同義語が 1 個でも使用されたもので，想定していた文脈同義語が全て使用されていないものを含む．15 対の同義語について，平均精度の変化を調べたところ，正規化しない場合の 8.4% から，正規化した場合は 43.0% に上昇した．その内容は，平均精度が上昇したものが 28 個，精度の低下したものが 2 個であった[6]．精度の低下した例は，クエリ「Final」に対する「最終」とクエリ「最終」に対する「Final」である．両例とも精度の低下した理由は同様なので，クエリ「Final」に対する「最終」の例を見てみると，正規化しない場合の順位 203 位から，正規化した場合は 355 位に落ちていた．理由としては，文脈同義語が想定していた 3 個の内，「搭載」⇒「Load」という 1 個しか使用されず，その文脈同義語も「Final」と「最終」を同義語として認識させ，他の候補語の順位は上昇させないようなものではなかったからであると推定される．

次に，文脈同義語のみ，文脈語の頻度制限（50～600）を外して実験を行った．その結果，想定していた全ての文脈同義語が使用され，25 対の同義語について平均精度が上昇したものが 46 個，低下したものが 2 個，変化しないものが 2 個であった．25 対の同義語について平均精度の変化を調べたところ，正規化しない場合の 9.6% から，正規化した場合は 42.9% に上昇した．

精度の低下した同義語の 1 つの例は，クエリ「GRP」に対する「グループ」である．類似度の値は，正規化しない場合の 0.24 から，正規化した場合の 0.32 に上昇していたが，他の候補語，

[6] 1 つの同義語対について，左辺から右辺と右辺から左辺で精度が異なるため，15 対の同義語では 30 個の精度計算が必要である．

例えば，「同行」の類似度が 0.21 から 0.39 のようにより上昇したために順位が低下したものである（「同行」は不正解である）．もう 1 つの低下した例は，クエリ「手順」に対する「PROC」であるが，その理由も同様である．

精度の変化しなかった同義語の 1 つ例は，クエリ「PROC」に対する「手順」である．正規化しない場合の「手順」の順位は 2 位で，1 位は「PROC」の同義語の「Procedure」であったが，正規化により双方の類似度が上昇し，結果として「手順」の順位は 2 位のままであった．もう 1 つの精度の変化しなかった例は，クエリ「DEP」に対する「出発」であるが，正規化しない場合の順位が 2 位はあり，その類似度は 0.42 であった．「遅延」⇒「Delay」という文脈同義語により正規化すると，「出発」の類似度は，0.48 に上昇したが，1 位であった「ARR」も同様に 0.48 から 0.57 に上昇した（「ARR」は不正解である）．これは，文脈同義語「遅延」⇒「Delay」が「出発」にも「ARR」にも関係しているためである．したがって，正解にのみ関係する文脈同義語を選択できれば，正解のみ精度を向上させることが期待できるが，どのように選択すればよいかは今後の課題である．

15 対の同義語の文脈語を正規化した場合の評価用辞書全体での平均精度は 45.7% であり，文脈語の正規化を行う前の 43.1% よりも向上していた．その理由の 1 つは，15 対の同義語の精度が上がり，その結果として平均精度を向上させたもの（その効果は，1.3%），もう 1 つの理由は，それ以外の同義語も文脈語の正規化により若干精度が向上したためである．

文脈同義語の頻度制限を外した 25 対の同義語の文脈語を正規化した場合の評価用辞書全体での平均精度は 46.8% であり，文脈語の正規化を行う前の 43.1% よりも向上していた．25 対の同義語の精度向上による効果は，2.1% であった．

結論として，文脈同義語は，第 4.3 節で述べたような高頻度と低頻度の文脈語の制限を外した方がよい結果となった．尚，頻度制限を外した場合，高頻度側で使用可能となった文脈同義語は 20 個，低頻度側で使用可能となった文脈同義語は 1 個であった．高頻度の文脈同義語が圧倒的に多かった．したがって，高頻度の同義語が既知である場合には，同義語の文字種に拘わらず，正規化することによりシステムの精度を向上できることが分かった．

5.3　大域的文脈情報との比較

酒井らは，日本語の略語からその原型語との対応関係を取得するのに以下のような手法を用いている (酒井，増山 2005)．

略語候補とそれに対応する原型語の候補を，それを構成している文字情報から獲得する．略語候補と原型語の候補の類似度を計算して，対応関係を取得する．

文脈情報の類似度について第 3 節で提案した手法との比較を行った．彼らは，漢字・ひらがなの名詞の略語を対象としたが，それをカタカナ，アルファベットに拡張して提案手法との比較を行った．彼らの類似度の計算は，コーパス中の略語候補語を含んでいる文書における略語候

表 8 大域的文脈情報と局所的文脈情報の比較

	平均精度（%）
酒井らの方式	7.4
window [2,2]	39.5

補語の出現頻度，全ての名詞の総出現頻度，文の数，略語候補語が最初に出現する文番号の情報を用いて重みを付与して順位付けを行い，その上位 N_n 文書を取り出して，略語候補の関連文書としている．次に，その関連文書に含まれる各名詞に対して出現頻度，文書頻度などの情報を用いて重みを付与して順位付けを行い，上位 N_m 個の名詞を取り出し，名詞の重みを付与したベクトルを生成している．原型語候補に対しても同様のベクトルを生成する．そして，その余弦により類似度を計算している．本論文でも酒井ら (酒井，増山 2005) と同様に，$N_n = 20$，$N_m = 200$ として実験した．

結果は，表 8 にあるように，提案手法よりも，かなり低い値となった．その原因として，略語とその原型語の対応関係を求めるのに，関連文書全体から代表的な名詞を抽出して類似度を計算している（大域的文脈情報）が，必ずしも，略語に関連する文書があるとは限らないと考えられる．我々は，局所的な文脈語から類似計算を行っている（局所的文脈情報）が，この手法の優秀性が証明された．

5.4 関連語の検索

語句には，同義であるもの以外に関連性のあるものが存在する．このような語句の分類も，テキスト処理においては重要である．例えば，「引き返し」という語句の関連語として「GTB (Ground Turn Back: 地上引き返し)」,「ATB（Air Turn Back: 空中引き返し)」,「RTO（Rejected Takeoff: 離陸中止)」,「トラブル」などがある．

「トラブル」は,「GTB」,「ATB」,「RTO」の上位概念あり,「GTB」,「ATB」,「RTO」は類義語である．これらの語句にも,「HYD Failure による GTB」,「HYD Failure による ATB」のように同じような文脈が現れる場合が多い．したがって，提案手法で関連語の検索も可能と考えられる．表 9 に，いくつかのクエリに対する回答の中での関連語を示す．

事実「RTO」について調べたところ，クエリに対する 50 位までの回答で類義語が 2 つ，原因を表す関連語が 5 つ，結果を表す関連語が 1 つ含まれていた．同じような文脈で使用される関連語は，本手法で検索できることが分かる．関連語によりどのように文脈が違うかについては．今後の研究課題である．

本論文では，精度は計算していないが，関連語の検索にも本手法が適用できると考えられる．

表 9　関連語の検索：番号は検索された番号，括弧内の「同」は同義語，「原」は原因を表す関連語，「結」は結果を表す関連語，括弧無しは類義語を示す．

クエリ	関連語
Trouble	1. TRBL（同）2. SQ（同）3. 不具合（同）5. トラブル（同）15. 問題　20. Surge（原）27. Wirirng（原）28. 故障（原）30. 事故（原）41. Burst（原）44. Leak（原）45. TCAS（原）50. Relay（原）
ATB	2. Divert　3. GTB　11. におい（原）19. RTO　23. Curfew（原）24. 引き返し　34. DVT　37. 遅延（結）49. WX（原）
GTB	1. RTO　6. ATB　10. ENG（原）14. 引き返し
RTO	1. 引き返し　2. GTB　4. Borescope（結）6. Flag（原）7. ATB　18. Warning（原）27. 喫煙（原）35. Alert（原）38. Brake（原）49. EGT（結）
引き返し	1. RTO　2. GTB　7. ATB　28. 喫煙（原）

6　複合名詞の処理

複合名詞の処理については，全ての連接する単名詞の組み合わせを調べると，その数が多くなり非効率である．したがって，最初に複合名詞を抽出して，それを単名詞と同様に扱うことで，これまで述べた処理と同じ手法を用いることとした．

複合名詞の抽出については，専門用語抽出システム (中川他 2003) を使用し，それが抽出したものの中で，重要度評価値が 3,000 以上の用語の中の複合名詞を使用した．専門用語抽出システムは，単名詞の左右に出現する単名詞の連接種類数と連接頻度および候補語の出現頻度から専門用語を抽出するものである．上記の条件で，350 の複合名詞が得られた．人手で複合名詞に対して同義語辞書を作成した結果，辞書の登録数 73 で，平均同義語数は 2.00 であった．複合名詞の同義語の中には，複合名詞と単名詞が含まれる．この複合名詞 350 と単名詞 1,343 に対して window [2,2] で文脈語の最小頻度 50，最大頻度 600 で文脈情報を取得した．実験の結果，平均精度は，44.3% であった．辞書登録数が少ないので単純には比較できないものの単名詞と同等の精度が得られた．

6.1　複合名詞と単名詞の関係

複合名詞と単名詞について以下のような関係があることが分かった．

(1)　複合名詞の同義語が単名詞の同義語の組み合わせでできているもの：
　　例：出発 遅れ - 出発 遅延

(2)　複合名詞の基底名詞[7] と単名詞が同義なもの：
　　例：搭乗 券 - 券

[7] 複合名詞を単名詞に分解した時に，複合名詞の最後に現れる単名詞．例の場合は，「券」．

(3) 複合名詞の基底名詞以外の語同士が同義なもの：
例：整備 点検 - 整備
(4) 複合名詞の中で一部の名詞に省略があるもの：
例：搭乗 旅客 数 - 搭乗 数
(5) 単名詞同士では，同義でなかったものが複合名詞では同義になるもの：
例：搭乗 口 - ゲート，到着 地 - 目的 地

以下では，複合名詞を最初に抽出した利点に着目して述べる．(1) については，複合名詞の処理を行わなくても，単名詞の同義語を置き換えることにより複合名詞の同義語を得ることが可能であるが，その場合には，「DEP 遅延」のようにあまり使用されない複合名詞の同義語が得られてしまい，単名詞の同義語を置き換えだけでは複合名詞の同義語を絞り込むことができない．したがって，複合名詞抽出の前処理を行うのが効率的である．(2) と (3) については，複合名詞を構成する名詞の中でより一般的で省略しても意味が変化しないものが省略されている．(4) の日本語の略語ついては，第 6.2 節で述べる．(5) の関係は，上記 4 種類と異なり，単純に省略や単名詞の置き換え，単名詞の同義語の組み合わせだけでは扱えないもので，複合名詞の前処理を行わないと同義語が得られないものである．

6.2 日本語の略語

日本語の略語の平均精度について調査した．日本語の略語とは，例えば，「整備作業」と「整備」のようなものであり，略語が原型語に完全に包含されるものである．したがって，「整備作業」と「整備点検」のようなものは含まれない．単名詞と複合名詞を合わせた同義語候補語 1,693 個について日本語の省略語の辞書を人手で作成したところ，エントリー数：92，項目数：123，平均項目数：1.34 であった．この辞書を使用して実験したところ，平均精度で 52.3% という高い精度が得られた．これは，日本語の原型語と一部省略されている略語では，その周辺には同じような文脈語が出現しやすいと考えられ，本手法の得意な分野だといえる．

7 同義語辞書作成

同義語辞書は，表 10 のように見出し語に対して 1 語以上の同義語が辞書項目として登録される．情報検索やテキストマイニングでは，同じ概念をグループ化し精度を向上させるために見出し語に対して 1 対 1 で同義語を対応させる必要がある場合がある．例えば，表 10 に対して，同義語リストは表 11 のようになる．表 11 では，「APP」が「進入」に，「Approach」が「進入」に，「CRZ」が「巡航」に変換されることを示す．「進入」，「巡航」は，変換されないでそのまま使用される．複数の同義語の中からどの語を変換語に選択するかは，専門用語抽出シス

表 10　同義語辞書

見出し語	登録語	
APP	Approach	進入
Approach	APP	進入
進入	APP	Approach
CRZ	巡航	
巡航	CRZ	

表 11　同義語リスト

見出し語	変換語
APP	進入
Approach	進入
CRZ	巡航

テムの重要度評価値の最も大きなものを用いた．つまり，同義語同士の中で最も重要度の高い語に変換するものである．また，「CRZ」⇒「巡航」と「巡航」⇒「CRZ」の場合には，「CRZ」と「巡航」の重要度評価値の大きな方に変換した．この例では，「巡航」の方が重要度評価値が大きく「CRZ」⇒「巡航」という云い換えになる．もちろん多義性のある語では，一意に同義語を決定できないのでこのようなリストは使用できない．この場合には，個々の語が出ている文脈から判断する必要があるが，これは今後の課題である．

次に同義語辞書を作成する際に一度に全て作成するのではなく，以下に示すように同義語辞書を一部作成した段階で同義語リストを文脈情報の正規化に使用するために文脈同義語としてシステムに与えることにより，残りの辞書作成の精度（平均精度）が向上するかを検証した．例えば，「PAX」を「旅客」に変換することにより，「PAX Boarding」と「旅客搭乗」の例では，「Boarding」と「搭乗」という同義語の文脈語が同一になる．第 5.2 節では，筆者の一人が選択した同義語対について，頻出する 1～4 個の文脈同義語を選択して使用したが，今回は出現頻度順に自動的にシステムに付与した．

文脈同義語は，第 5.2 節の文脈語の正規化で頻度制限をしない方が精度が良かったので，本節でも頻度制限を行わなかった．出現頻度が 500 以上の同義語リストを作成したところ，41 個の同義語リストが得られた．これを文脈同義語とし，正規化したものとしないものについて平均精度を比較したところ，それぞれ平均精度は 43.3%，41.1% であり，正規化したものの方が約 2.2% 精度が向上した（文脈同義語リストに含まれる同義語は評価から除外した）．同様に出現頻度が 300 以上，1,000 以上のものについて比較したところ，正規化したものの平均精度がそれぞれ約 2.2%，1.8% 精度が向上した．出現頻度が 100～300，300～500 のものについては，平均精度の変化はなかった．以上の結果から，同義語の中で頻度の高いものを文脈同義語としてシステムに付与すると，若干平均精度が向上することが分かった．

8 結論および今後の課題

本論文では，特定分野における同義語辞書作成支援システムを提案した．提案手法は，語句の境界が認識できればよいので，深い言語処理技術は必要とせず，分野・言語を問わないものである．クエリ・候補語の前後に出現する語句の文脈情報のみを使用したが，人間の辞書支援システムとしては，十分に機能することを実験の結果確認した．文字種が異なり精度の低い同義語については，文脈語を正規化することにより，精度を向上できる事を確認した．実験の結果，以下の知見が得られた：

- window 幅による精度の比較では，Terada ら (Terada et al. 2004) の英文の略語を対象としたものでは window [3,3] が最も精度が良かったと報告されているが，本論文で使用した日本語のコーパスでは，window [2,2] が最も精度がよかった．日本語の漢字 1 文字の持つ情報量は，明らかに英語 1 文字の情報量よりも多いので，漢字を含む日本語のコーパスの方が英文のコーパスよりも window 幅が小さい場合に精度がよい事が，文脈情報という形で確認できた．
- 同義語の字種別での平均精度は，アルファベット同士が最も高く，カタカナとアルファベットの平均精度は，最も低かったが，その原因は周辺の文脈情報の文字種が異なる場合が多いからである．
- 文脈語の正規化について，以下の 2 点で，その有効性を確認できた．
  1 番目は，いくつかの同義語対について文脈同義語をシステムに与えることにより，その同義語対の精度をかなり向上させることができた．その理由は，文脈語の正規化をすることにより，異なる文字種の同義語の周辺に出現する異なる文字種の文脈同義語が同定されるためである．また，PortableKiwi を用いることにより，対象とする同義語の周辺に頻出する文脈語を簡単に選択できることが分かった．
  2 番目は，同義語辞書の作成途中で，出現頻度の高い同義語をシステムに与えることにより，システムの精度が向上した．したがって，同義語辞書を出現頻度の高いものから作成し，作成した同義語を知識としてシステムに与えることにより，それ以降の同義語抽出の精度を向上できることが分かった．

今後の課題としては，以下が挙げられる：

- 文脈同義語として，PortableKiwi を用いて候補語に特徴的な語句を選択して，本システムとハイブリッドに使用できるようにすると，精度を向上できる可能性がある．
- 航空分野だけでなく他の分野の同義語でも本手法をテストして有効性を確認する必要がある．
- 多義語の処理については，クエリに対する典型的な文脈（ベクトル）情報が得られていれば，そのクエリが出現する文脈から多義性を解消できる可能性がある．例えば，「Noise

について Cabin に問い合わせたところ，Cabin での Noise は，Door 近くからであることが判明した」という文の 1 番目の Cabin は，客室乗務員のことであり，2 番目の Cabin は，客室のことである．

- 同義語の辞書作成というテーマで議論したが，語の意味的な関係は複雑であり，同義語の中にはある場面では同義語であるが，別の面では上位—下位概念として扱わないといけないものがある．例えば，「引き返し」と「ATB（空中引き返し）」，「GTB（地上引き返し）」において，「引き返し」は「ATB」，「GTB」の上位概念である．「引き返し」に関する事例を収集したい場合には，「引き返し」を「ATB」，「GTB」と同義で扱ってよいが，更に詳細に分類したい場合には，上位—下位概念として扱わなければならない．したがって，今後オントロジーを構築する手法についても研究する必要がある．

謝　辞

専門用語自動抽出システムは，東京大学中川研究室・横浜国立大学森研究室で開発された用語抽出システムを使用させて頂きました．ここに感謝いたします．

参考文献

Knight, K. and Graehl, J. (1998). “Machine Transliteration.” *Computational Linguistics*, **24** (4), pp. 599–612.

Masuyama, T. and Nakagawa, H. (2005). “Web-based aquisition of Japanese katakana variants.” In *SIGIR '05: Proceedings of the 28th annual international ACM SIGIR conference*, pp. 338–344.

Ohtake, K. and Sekiguchi, Y. (2004). “Detecting Transliterated Orthographic Variants via Two Similarity Metrics.” In *Proceedings of Coling 2004*, pp. 709–715.

Pustejovsky, J., Castao, J., Cochran, B., Kotecki, M., Morrell, M., and Rumshisky, A. (2001). “Extraction and Disambiguation of Acronym-Meaning Pairs in Medline, unpublished manuscript.”.

Schwartz, A. S. and Hearst, M. A. (2003). “A Simple Algorithm for Identifying Abbreviation Definitions in Biomedical Text.” In *Pacific Symposium on Biocomputing*, pp. 8: 451–462.

Tanaka-Ishii, K. and Nakagawa, H. (2005). “A Multilingual Usage Consultation Tool based on Internet Searching—More than search engine, Less than QA.” In *The 14th International World Wide Web Conference (WWW2005)*, pp. 363–371.

Terada, A., Tokunaga, T., and Tanaka, H. (2004). “Automatic expansion of abbreviations by

using context and character information." *Inf. Process. Manage.*, **40** (1), pp. 31–45.

阿玉泰宗，橋本泰一，徳永健伸，田中穂積 (2004). "日英言語横断情報検索のための翻訳知識の獲得." 情報処理学会論文誌：データベース, **45** (SIG 10), pp. 37–48.

酒井浩之，増山繁 (2005). "略語とその原型語との対応関係のコーパスからの自動獲得手法の改良." 自然言語処理, **12** (5), pp. 207–231.

寺田昭，吉田稔，中川裕志 (2006). "文脈情報による同義語辞書作成支援ツール." 情報処理学会研究報告, pp. 87–94.

寺田昭，吉田稔，中川裕志 (2007). "同義語の類似度に関する考察." 言語処理学会第 13 回年次大会, pp. 1097–1100.

中川裕志，森辰則，湯本紘彰 (2003). "出現頻度と連接頻度に基づく専門用語抽出." 自然言語処理, **10** (1), pp. 27–45.

藤本宏凉，吉田稔，中川裕志 (2005). "ローカルコーパスからのテキストマイニングツール：PortableKiwi." 言語処理学会第 11 回年次大会, pp. 97–100.

略歴

寺田　昭：1976 年京都大学工学部電気工学第二学科卒業．1978 年同大学大学院工学研究科修士課程電気工学第二専攻修了．2003 年東京工業大学大学院情報理工学研究科後期博士課程修了．博士（工学）．現在，（株）日本航空インターナショナル勤務．自然言語処理，テキストマイニング，航空安全に興味を持つ．言語処理学会会員．

吉田　稔：1998 年東京大学理学部情報科学科卒業．2003 年東京大学大学院理学系研究科情報科学専攻博士課程修了．博士（理学）．2003 年より東京大学情報基盤センター図書館電子化研究部門助手．2007 年より同助教．自然言語処理，Web 文書解析の研究に従事．

中川　裕志：1975 年東京大学工学部卒業．1980 年東京大学大学院工理学系研究科修了．工学博士．同年より横浜国立大学工学部勤務．1999 年より東京大学情報基盤センター教授．現在に至る．2000 年から 2002 年言語処理学会編集長，2002 年から 2004 年言語処理学会総編集長，2004 年から 2006 年言語処理学会会長，2006 年より情報処理学会自然言語処理研究会主査．自然言語処理，機械学習，WWW の研究に従事．

(2007 年 7 月 31 日　受付)
(2007 年 11 月 20 日　再受付)
(2007 年 12 月 25 日　採録)