

係り受けの 2 部グラフと共起関係を利用した同義表現抽出

上野 友司[†] 森 辰則[‡] 木戸 冬子^{††} 中川 裕志^{‡‡}

[†] 横浜国立大学大学院 環境情報学府
tomot@forest.eis.ynu.ac.jp

[‡] 横浜国立大学大学院 環境情報研究院
mori@forest.eis.ynu.ac.jp

^{††} マイクロソフト株式会社
fkido@microsoft.com

^{‡‡} 東京大学 情報基盤センター
nakagawa@dl.itc.u-tokyo.ac.jp

概要

近年、文書の有効利用を目的として大量の文書データを様々な観点から分析し、役立つ情報を抽出するテキストマイニングが注目を集めている。テキストマイニングの基本機能の一つに概念の抽出があるが、そこでは語の多義性の解消とともに異表記同義語の認定が重要となる。

そこで、本稿ではその際に必要な同義語辞書作成の支援となる、同義表現の候補を多量の文書群から抽出する手法を提案する。具体的には係り受けの 2 部グラフから可能性のある候補を完全 2 部グラフを用いて大まかに抽出し、その候補間における表現の近さの度合を、語の前後に出現する文脈の情報を用いて類似度として計算する。さらにその類似度を元の文脈情報にフィードバックさせることで、候補抽出の適合率、再現率を上げる手法も検討する。

A Method for Extraction of Similar Expression using Bipartite Graph of Word Dependency and Co-occurrences

Yuji Ueno[†] Tatsunori Mori[†] Fuyuko Kido[‡] Hiroshi Nakagawa^{††}

[†]Graduate School of Environment and Information Sciences, Yokohama National University
{tomot,mori}@forest.eis.ynu.ac.jp

[‡]Microsoft Corporation
fkido@microsoft.com

^{††}Information Technology Center, The University of Tokyo
nakagawa@dl.itc.u-tokyo.ac.jp

Abstract

Text Mining is attracting a great deal of attention because it is a tool to extract useful information from a large amount of documents in terms of various viewpoints. One of basic functions of Text Mining is the extraction of important concepts. In the process, it is important to recognize different expressions with a same meaning as well as to perform the word sense disambiguation.

In this paper, we propose a technique to extract candidates of synonyms from a large amount of documents. It supports to making synonym dictionary which is used maintaining the consistency of the expressions. Firstly, a set of candidates of synonyms are roughly extracted using a complete bipartite graph of word dependency. Secondly, degree the numerical value of similarity between the candidates by using information on the context which appears before and behind the word. In addition, we examine the extraction technique to rise the precision and the recall of candidates by feeding back the numerical value to former context information.

1 はじめに

現在、企業活動の活性化を目的としたテキストマイニング [1] が注目を集めている。企業で蓄積している情報の中でも文書形式で保存されている情報の量は非常に多い。そのため、大量の文書データを様々な観点から分析し、役立つ情報を抽出するテキストマイニングを利用することによって、業務効率が向上することが期待されている。

テキストマイニングの基本機能の一つに概念の抽出があるが、そこでは語の多義性の解消とともに異表記同義語の認定が重要となる。例えば、テキストマイニングの基本ツールとして情報検索システムがよく用いられているが、そこでは、利用者の入力する検索質問中の表現と文書に実際に登場する表現との相違が検索効率に影響を与える。利用者が「A をインストールしたい」と尋ねたときに、文書中にある「A をセットアップする方法」を検索することは容易ではなく、利用者が順次言い換えを行っていくのが現実である。特に専門用語が多数登場する分野においては、同じ事物でもコミュニティによって専門用語の表層表現が違う場合や、筆者によって異なる略語を用いるケースが多い。このような異表記同義語の認定を行うためには、一般的な同義語辞書を利用するだけでは不十分であり、その分野に特有の同義表現辞書を用意することが必要となる。

本稿では、上記のような限定された分野の同義語辞書を一から作成する際の支援を目的として、同義表現の候補をその分野に属する大量の文書から抽出する手法を示す。

2 関連研究

同義表現抽出に関連する研究としては類義語や換言知識を自動抽出するものがあり、それらは近年盛んに検討されている。村上 [2] らは各々の名詞に対して他の単語との係り受け関係を頻度とともに情報ベクトルとして得、そのベクトルの内積を用いて名詞間における同義語の候補を抽出する方法を提案している。その際、候補として誤って取り出されてしまった反意語等のノイズを除去するために筆者ごとの表現の違いを利用している。中渡瀬 [3] は複合語内の名詞に注目して名詞間の接続関係を表現する 2 部グラフを作成し、そのうちで極大完全 2 部グラフを形成している部分を見つけることで類義語を抽出している。また仁井 [4] らは動詞に係る「名詞 + 表層格」の間の類似度計算を行い動詞間における換言知識の抽出を試みている。また山本 [5] は各内容語の係り受け関係の関係子とその頻度を集計し、内容語間における係り受け関係の出現確率を用いて換言可能性の算出を行っている。

本研究では抽出の対象を名詞や動詞などに限定せず、基本的に全ての品詞を対象としている。なぜならば本研究の目的である同義語辞書作成の支援の際には、分野固有の名詞はもちろんのこと、それらの名詞と関連して出現する分野固有の動詞や形容詞、形容動詞なども網羅する必要があるからである。よって様々

な品詞が混在する中で偏りなく同義表現を抽出する手法が必要となる。

3 手法と実装

本章では同義表現抽出の具体的な手法を述べる。

3.1 処理の概要

本研究では入力文書に対しいくつかの処理を行って同義表現の候補を抽出し、その候補の表現の近さの度合を類似度として算出する。その類似度をもとに候補の抽出を行う。さらに抽出結果を類似度計算ヘフィードバックさせることで精度の向上を試みる。それらの処理の流れを図 1 に示す。

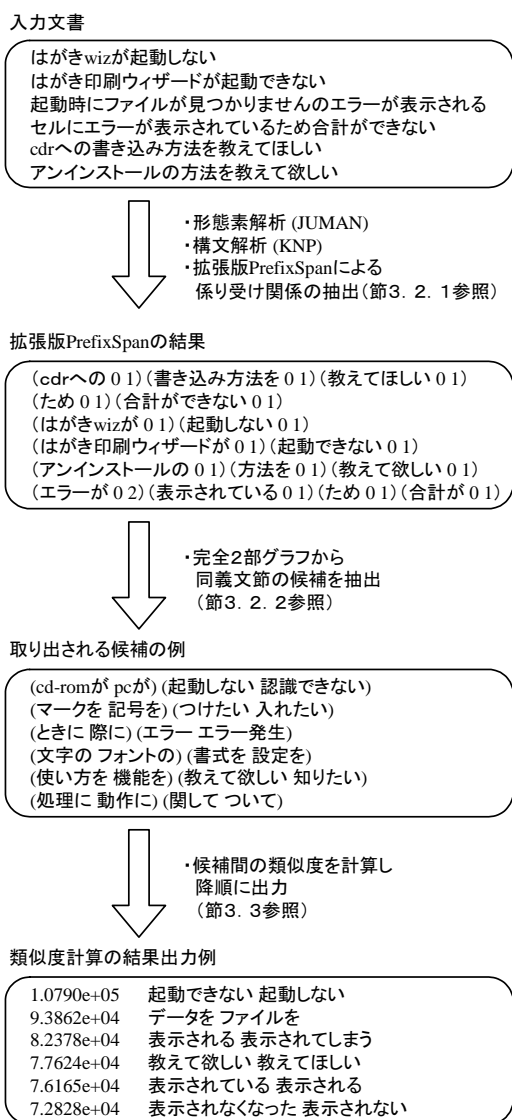


図 1: 同義表現の候補抽出および類似度計算

3.2 完全2部グラフを用いた同義表現の候補抽出

まずはじめにテキストから文節間の係り受け関係を抽出する。この作業を効率的に行うために拡張版 PrefixSpan[6]を用いる。

3.2.1 拡張版 PrefixSpan

PrefixSpanとはシーケンシャルパターンマイニングで用いられるアルゴリズムである。自然言語処理におけるシーケンシャルパターンマイニングの最も単純な例は、テキスト中に ξ 回以上出現する単語の列を漏れなく抽出することであるが、PrefixSpanアルゴリズムはこれを効率的に行うものである。

これに係り関数を導入した拡張版 PrefixSpan[6]では、係り受け関係を考慮した頻出パターンを抽出することができる。係り受けを考慮した拡張版 PrefixSpanの出力結果の一部を図2に示す。なお、このプログラムは構文解析した結果を入力とし、形態素解析器にはJUMAN[7]を、構文解析器にはKNP[8]を用いた。

```
(画像を 0 3)(表示させる 0 3)(方法は 0 3)
(綺麗な 0 2)(正円を 0 2)(描きたい 0 2)
(綺麗に 0 8)(画像を 1 3)(表示させる 0 3)
(綺麗に 0 8)(背景が 1 5)(表示されない 0 5)
(綺麗に 0 8)(表示させる 0 3)(方法は 0 3)
(正円を 0 2)(描きたい 0 2)
(背景が 0 5)(表示されない 0 5)
(表示させる 0 3)(方法は 0 3)
```

図2: 拡張版 PrefixSpan 出力例

拡張版 PrefixSpanの出力は(アイテム 関係 頻度)が係り受け順に後ろに連なった形となっている。ここで「関係」とは表層的にアイテムの直後にある文節と、アイテムに係っている文節との係り受け上の距離である。この出力結果において各行の前から n 個の要素を集計することで、全文書中に出現している n 接続の文節、そのすべての係り受けの種類、およびその頻度を求めることができる。

なお PrefixSpanの前処理として、それ単体では具体的な述部を構成しない

- できる
- 行う
- 出る
- ある
- なる

などの動詞およびその活用形は、表層表現上でこれらの直前に出現している文節とまとめたものを一つの文節として取り扱う。

3.2.2 完全2部グラフ

拡張版 PrefixSpanの出力結果からテキスト中に存在するすべての文節間の係り受け関係が得られる。その情報を用いて各文節を頂点、係り受け関係を辺とする2部グラフを作成することができる。作成される2部グラフの例を図3に示す。

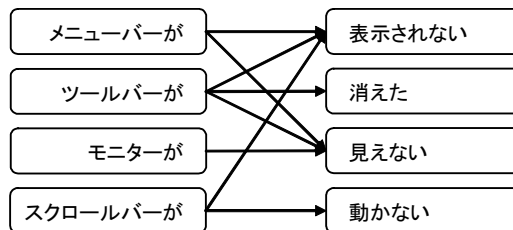


図3: 係り受け関係の2部グラフ

この2部グラフ中で完全2部グラフを形成しているところに注目してみる。完全2部グラフとは2部グラフの頂点集合を X 、 Y とするとき X (または Y)の任意の点が Y (または X)の全ての点と接続されているものである。図3において完全2部グラフとなっているのは図4に示す部分である。

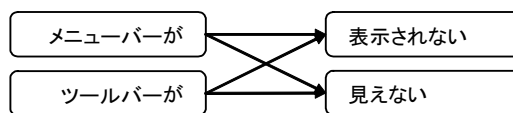


図4: 完全2部グラフ1

完全2部グラフを形成している頂点集合は互いに共通の要素と係り受け関係を持っているため、左側(始頂点集合)および右側(終頂点集合)の文節組それぞれが言い換えができる可能性のある文節の組、すなわち、同義文節の候補の組となる。図4の場合においては(メニューバーが ツールバーが)の組、(表示されない 見えない)の組がそれぞれ同義文節の候補の組にあたる。これらの始頂点集合、終頂点集合を同義表現の候補として2部グラフから抽出していく。

ただし係り受けの関係においては、ある1つの文節に対して複数の文節に係る場合があるため、それによって作られる完全2部グラフが多数存在することが考えられる。例えば

- excelをPCにセットアップしたい
- excelをPCにインストールしたい

という2つの文がある。これらの文を構文解析すると図5のようになる。

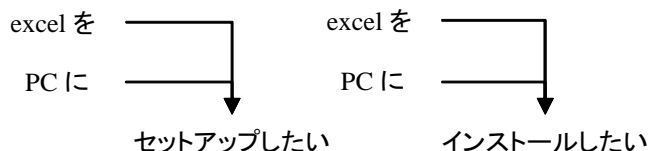


図5: 構文解析結果

この結果から2部グラフを作成すると図6の形になる。

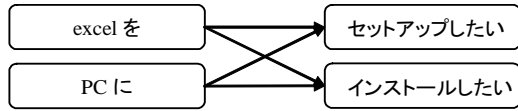


図 6: 完全 2 部グラフ 2

図 6 は完全 2 部グラフを形成しているので (excel を PC に) (セットアップしたい インストールしたい) が同義文節の候補となってしまふ。この時、(excel を PC に) は同義文節の組となっているとはいい難い。これは係り受け関係をすべて均一に扱っているために生じる。そこで完全 2 部グラフを探索する際に係り受けの種類 (格、接続形態など) も考慮し、同じ係り受けの種類を持つもの同士を頂点集合とするようにする。

図 7 に完全 2 部グラフによって抽出された、同義表現の候補の一部を示す。

(幅を 太さを) (変えたい 変更したい)
 (動作が 動きが) (遅くなる 悪くなる)
 (動作が 動きが) (遅くなる 遅くなった)
 (cdrom を cd-rom を) (挿入しても 入れても)
 (パソコンに マシンに) (移したい 入れたい)
 (パソコンに pc に) (導入して インストールして)
 (メニューバーが ツールバーが) (無くなった 消えた)
 (メニューバーが ツールバーが) (無くなった なくなった)
 (メニューバーが ツールバーが) (でない でてこない)
 (setup が セットアップが) (行えない 進まない)
 (setup が セットアップが) (終了しない 進まない)
 (強制終了が エラー msg が) (起こる 発生する)

図 7: 同義表現の候補

完全 2 部グラフを形成する頂点集合のサイズ (位数) は 2 以上であるので、完全 2 部グラフの最小部分は $(a\ b)(x\ y)$ という 2-2 の形である。 $(a\ b)(x\ y)$ と $(a\ b)(y\ z)$ という 2 つの完全 2 部グラフがあった場合、これらを併合することで $(a\ b)(x\ y\ z)$ という 1 つの完全 2 部グラフにまとめることができるが、本稿では 3.3 節で行う類似度計算を考慮して頂点集合の位数 2 のものを順時出力していく形式を取っている。

3.3 文脈の共起情報を用いた類似度の計算

3.2 節で同義表現の候補となる文節の組 (候補文節の組) が抽出された。本節では候補文節に係る、もしくは候補文節が係っている文節郡の頻度情報を用いて類似度を算出し候補間の順位付けを行う手法を述べる。候補として $(a\ b)(c\ d)$ が抽出された場合、 a と b 、 c と d の類似度を計算し降順に出力させる。

類似度を計算するにあたり、ある文節 a に注目したとき、その文節 a に対して直接係る文節 pre_i の集合を前文脈、 a が直接係る文節 $post_j$ の集合を後文脈と考え、文脈中の各文節の重みを成分にした次のベクトルを考える。

$$CV_a = \{w_{pre_1}^a, \dots, w_{pre_m}^a, w_{post_1}^a, \dots, w_{post_n}^a\} \quad (1)$$

文節間の文脈の類似度は対応するベクトルの内積により求めることとする。

$$sim(a, b) = CV_a \cdot CV_b \quad (2)$$

上記ベクトルの各成分の値は次のように決めた。

$$w_{pre_i}^a = tf(a, pre_i)icf(pre_i) \quad (3)$$

$$w_{post_j}^a = tf(a, post_j)icf(post_j) \quad (4)$$

$$tf(a, pre_i) = \log_2(freq_{pre_i}^a) + 1 \quad (5)$$

$$tf(a, post_j) = \log_2(freq_{post_j}^a) + 1 \quad (6)$$

$$icf(pre_i) = \log_2 \frac{pre_N}{cf(pre_i)} \quad (7)$$

$$icf(post_j) = \log_2 \frac{post_N}{cf(post_j)} \quad (8)$$

ただし

$freq_{pre_i}^a$: 文節 a に係っている文節 pre_i の出現頻度

$freq_{post_j}^a$: 文節 a が係っている文節 $post_j$ の出現頻度

$cf(pre_i)$: 文節 pre_i が前文脈に出現する頻度

$cf(post_j)$: 文節 $post_j$ が後文脈に出現する頻度

pre_N : 全前文脈数

$post_N$: 全後文脈数

式 (7)、(8) で示した icf は idf (Inverse Document Frequency) の文書 (Document) を文脈 (Context) に置き換えたものである。係り受け関係の頻度における、ある文節が出現している割合の逆数で求める。つまり多くの文節の文脈に出現している文節ほどその値は小さくなり、逆に少ないと値は大きくなる。本研究のように全品詞の係り受け関係からベクトルを作成した場合、多種類の文節が係り易い「こと」「方法」などの名詞、同じ対象に対する複数の動作表現 (「ファイルを」に対する「コピーする」「削除する」「消す」「作成する」「張り付ける」など) が多数存在し、またその頻度も大きい。これらの係り受けし易い文節は特定の文脈を判別する時の手がかりとはならないので、その頻度の影響を抑える指標として icf を用いた。

3.4 類似度計算における文脈となる文節の制限

文節によっては比較的ゆるやかな係り受け関係しか構成しないものもある。そのため類似度計算に用いる文脈に制限をかけることで、抽出の精度が向上することが期待される。ここでは一つの節中における係り受け関係よりも、節を越えた係り受けを重視することを行った。具体的に「名詞 + の」や、副詞、形容詞などを文脈から除いて、動詞ならびに「名詞 + ノ以外の格助詞」で限定したものを文脈として扱うことにする。

3.5 類似度計算の結果を用いたフィードバック

類似度計算によって文節間の類似度が数値として表される。この数値が高く算出された文節の組を同義文節として同一視し、一連の計算を再度行うことで抽出の精度を上げる手法がいくつか考えられる。

4.3 完全2部グラフを用いた同義表現の候補抽出

完全2部グラフを構成した文節組の候補となるものは全部で81319組であった。類似度計算では最も精度が高くなると考えられる「文脈に制限を加えた類似度計算」の手法を採用し、前述の評価基準に従って判別した。それぞれの評価1と評価2の例を下記に示す。

(起動できない 起動しない)
 (知りたい 教えて欲しい)
 (関して ついて)
 (セットアップしたい インストールしたい)
 (強制終了 フリーズ)
 (マシンに PCに)
 (無くなった 消えた)
 (使用する 使う)
 (winxpに xpに)
 (良いのか よいのか)

図 10: 評価1と判定された文節組の例

評価1と判定されたもので特に上位には、(教えて欲しい 教えてほしい)などの漢字/かなの表現の差異や(表示される 表示されてしまう)などの補助動詞の差異による文節組が多数見られた。

(開くと クリックすると)
 (表示される 要求される)
 (画像を 図を)
 (データが ファイルが)
 (変更する 切り替える)
 (間隔を 行間を)
 (方法を 手順を)
 (ツールバーが ime ツールバーが)
 (値を 文字を)
 (インポートしたい 取り込みたい)

図 11: 評価2と判定された文節組の例

評価2と判定されたものには(画像を ファイルを)などのようにis-a関係にある組が多く見られた。これらは前後に出現する文脈によって対象の範囲が限定されることで、同義のものと同様の働きを持つものだと考えられる。

また評価3の中には4.2で述べた類義語、反意語がいくつか見られた。これらの例を図12と図13に示す。

(コピーしたい 張り付けたい)
 (フォルダを ファイルを)
 (fdに cd-rに)
 (図を 表を)
 (文字の 数字の)
 (faxの メールの)

図 12: 類義語の例

(表示したい 消したい)
 (起動できない 終了できない)
 (表示したい 非表示にしたい)
 (開く 閉じる)
 (削除したい 保存したい)

図 13: 反意語の例

前後に出現する文脈の情報だけで類似度を計算しているため、どうしても類義語や反意語、固有名詞を含む名詞句などは類似度が高くなる。本稿では「どんな文脈でも同義となり得ない」と評価したが、一方で、これらは有用な知識になることがある。例えば「~を表示したい」といった要求があった場合には、その関連知識として対になる操作である「~を消す」のための方法や手段も併せて提示すると利用者の理解を一層深めることができると考えられる。

取り出した文節組の件数とその精度の変化を図14に示す。

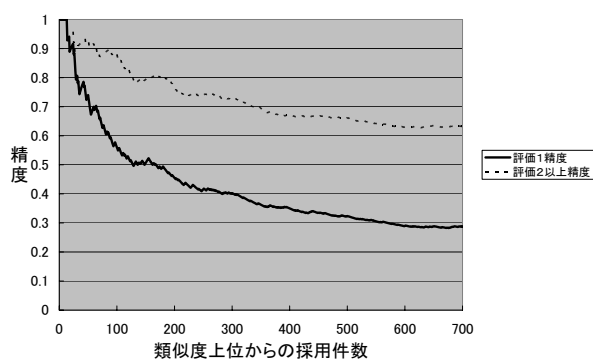


図 14: 抽出精度

上位の200組においては文脈に依存せずと同義といえる文節組が45.50%、文脈によって同義といえる文節組も合わせると76.50%の精度となった。

4.4 類似度計算の結果を元にしたフィードバック

この節では3.5節で提案したフィードバックによる影響を評価してみる。フィードバックをかけたことにより類似度の上位700単語組に新しく加わった文節組を4.3節と同様の手法で評価し、上位700以内の評価の構成比の変化を求めた。

4.4.1 上位語の併合

上位語の併合によって上位700以内の構成比の変化を表1に示す。件数はそれぞれの評価ごとの件数の差、構成比の差は繰り返し計算(10回)前と後のそれぞれの評価が占めるパーセンテージの差で求めている。

表 1: 上位語の併合による構成比の変化

	新規追加単語組数 14		
	評価 1	評価 2	評価 3
件数 (件)	+1	+4	-5
構成比の差 (ポイント)	+0.2	+0.6	-0.8

上位語の併合は適合率を上げる処理として採用したものである。700 件に対して 14 件の追加ということで変化の割合は小さいが、評価 1・評価 2 の件数が増加し評価 3 の件数は減少したので、ある程度の効果が見られている。

4.4.2 新しい完全 2 部グラフの発見

新しい完全 2 部グラフの発見によって上位 700 以内の構成比の変化を表 2 に示す。

表 2: 新しい完全 2 部グラフの発見による構成比の変化

	新規追加単語組数 27		
	評価 1	評価 2	評価 3
件数 (件)	-3	+4	-1
構成比の差 (ポイント)	-0.4	+0.6	-0.2

再現率を上げる手法として新しい 2 部グラフを発見する処理を行った。評価 2 の件数は増加し評価 3 の件数は減少したが、評価 1 の件数が減ってしまった。この手法では実際の文書に現れない係り受け関係を仮想的に生成しているわけだが、その精度が低いのかもれない。より詳細な検討が必要である。

5 まとめと今後の課題

本稿では、特定分野に属する文書群から完全 2 部グラフを用いて同義表現となる文節を大まかに抽出し、前後に出現する文脈の情報を用いて表現の近さの度合を類似度として求める手法を提案した。さらにその上位の結果を一連の処理にフィードバックし抽出の精度を上げる手法も提案した。完全 2 部グラフの構成から抽出した同義表現の候補にはノイズも数多く残っているが、指標 *tf·icf* を用いて文脈の類似度を求めることで、ランクの上位から 200 件においては比較的近い表現といえるものが 76.50% の精度で取り出すことができる。これによって一般的な同義表現および特定の文脈を持つことで同義表現になり得るものを容易に集めることができ、同義語の集合を求める際の支援として適当だと考えられる。また類似度が高く算出された文節組を同義文節として同一視し、各処理にフィードバックさせ抽出の精度を向上させる手法も検討した。上位語を併合させ類似度を再計算する手法と新しい完全 2 部グラフを発見することで同義文節の候補を追加させる手法を検討した。両手法とも抽出精度を向上

させてはいたが、その効果は小さいという結果となった。

今後の課題としては、まず係り受けの情報を細分化し類似度計算に反映させる手法がある。本稿では単純な係り受け関係のみを用いて計算を行っていたが、節を越えた係り受け関係などを分類したりすることでより細かい抽出が実現できると考えられる。また本稿では文脈に制限をかけて類似度計算を行ったが、候補語にも制限をかけることで抽出の目的に沿った同義語を収集できるようになると考えられる。

6 謝辞

拡張版 PrefixSpan のプログラムを提供して頂いた奈良先端科学技術大学院大学の工藤拓さんに深く感謝致します。

参考文献

- [1] 市村由美, 長谷川隆明, 渡部勇, 佐藤光弘. テキストマイニング - 事例紹介. 人工知能学会誌, Vol. 16, No. 2, pp. 192-200, 3 月 2001.
- [2] 村上明子, 那須川哲哉. 複数の筆者の表記の違いを利用した同義語抽出の精度向上. 言語処理学会第 8 回年次大会発表論文集 A1-5, 言語処理学会, 3 月 2002.
- [3] 中渡瀬秀一. 複合語からの類義語抽出法. 情報処理学会デジタル・ドキュメント研究会, Vol. 32, No. 6, pp. 39-46, 3 月 2002.
- [4] 仁井康夫, 酒井浩之, 吉田辰巳, 増山繁. 動詞間の換言知識の自動獲得. 言語処理学会第 9 回年次大会発表論文集 B3-2, 言語処理学会, 3 月 2003.
- [5] 山本和英. テキストからの語彙的換言知識の獲得. 言語処理学会第 8 回年次大会発表論文集 A6-5, 言語処理学会, 3 月 2002.
- [6] 工藤拓, 山本薫, 坪井祐太, 松本裕治. 言語情報を利用したテキストマイニング. 情報処理学会自然言語処理研究会, Vol. 148, No. 10, pp. 65-72, 3 月 2002.
- [7] 黒橋禎夫, 長尾真. 日本語形態素解析システム JUMAN version 3.6 使用説明書. 京都大学大学院 情報学研究科, 1998.
- [8] 黒橋禎夫. 日本語構文解析システム KNP version 2.0b6 使用説明書. 京都大学大学院 情報学研究科, 1998.

付録：類似度上位の結果

順位	文節 a	文節 b	評価
1	起動できない	起動しない	1
2	教えて欲しい	教えてほしい	1
3	表示される	表示されてしまう	1
4	データを	ファイルを	1
5	知りたい	教えて欲しい	1
6	表示されなくなった	表示されない	1
7	表示されている	表示される	1
8	表示されていない	表示されない	1
9	知りたい	教えてほしい	1
10	作成する	作成したい	1
11	いいのか	よいか	1
12	分からない	わからない	1
13	強制終了	エラー	1
14	"msg	msg	-
15	文書を	ファイルを	1
16	エラー	エラー msg	1
17	表示され	表示される	1
18	起動すると	開くと	2
19	開いて	ついて	1
20	開こうとすると	開くと	1
21	セットアップしたい	インストールしたい	1
22	無い	ない	1
23	変更したい	設定したい	1
24	入れても	挿入しても	1
25	印刷される	表示される	3
26	開かない	開けない	1
27	エラー	発生	3
28	印刷されない	表示されない	2
29	保存した	作成した	2
30	いいか	よいか	1
31	表示された	表示される	1
32	開くと	クリックすると	2
33	いいか	いいのか	1
34	画像を	ファイルを	2
35	msg で	エラーで	2
36	変更したい	変えたい	1
37	入力すると	入れると	1
38	削除したい	消したい	1
39	いいのか	よいか	1
40	いいのか	よいか	1
41	mail メールの	メールの	1
42	インストールしたい	インストールする	1
43	表示される	要求される	2
44	案内	ご案内	1
45	表を	ファイルを	2
46	インストールできない	起動できない	3
47	エラー	フリーズ	2
48	なくなってしまった	消えてしまった	1
49	なくなった	消えてしまった	1
50	消えてしまった	消えた	1
51	メールを	ファイルを	2
52	開くと	印刷すると	3
53	文字を	ファイルを	2
54	表示したい	出したい	2
55	データを	文字を	2
56	入力する	入力したい	1
57	作成したい	作りたい	1
58	強制終了	フリーズ	1
59	止まる	止まってしまう	1
60	起動しない	起動しなくなった	1

順位	文節 a	文節 b	評価
61	"	msg	-
62	時に	ときに	1
63	いいのか	いいか	1
64	変更する	設定する	1
65	作成したい	印刷したい	3
66	表示したい	入力したい	3
67	知りたい	確認したい	1
68	変更したい	削除したい	2
69	表の	セルの	2
70	文字の	表の	3
71	起動できない	起動しなくなった	1
72	戻したい	変更したい	2
73	起動できない	開けない	2
74	表示	発生	2
75	画像を	図を	2
76	表示したい	表示する	1
77	作った	作成した	1
78	挿入すると	開くと	2
79	データを	画像を	2
80	データを	文書を	2
81	入力すると	開くと	2
82	なくなってしまった	なくなった	1
83	ときに	際に	1
84	押すと	クリックすると	2
85	起動できない	起動できなくなった	1
86	文字を	表を	2
87	データが	ファイルが	2
88	追加したい	インストールしたい	2
89	使用できない	使えない	1
90	msg が	画面が	3
91	住所録を	ファイルを	3
92	強制終了	エラー msg	2
93	表示したい	消したい	3
94	文字が	画面が	2
95	起動すると	起動しようとする	1
96	入れたい	入力したい	1
97	消したい	非表示にしたい	1
98	入力した	作成した	2
99	挿入したい	入力したい	2
100	データを	表を	2
101	強制終了	発生	3
102	作成したい	削除したい	3
103	ことができるか	ことはできるか	1
104	エラー発生	エラー	1
105	メッセージ	エラー	2
106	よって	ついて	3
107	メールが	ファイルが	2
108	fax の	メールの	3
109	文字の	セルの	3
110	なくなった	消えた	1
111	表示されなくなってしまった	表示されない	1
112	画像を	表を	3
113	表の	テキストボックスの	3
114	マシンに	pc に	1
115	表示	エラー	3
116	フリーズする	発生する	2
117	作成したい	入力したい	2
118	なくなってしまった	消えた	1
119	表示されなくなった	表示されていない	1
120	作成したい	挿入したい	2