卒 業 研 究 報 告 題 目

テキストマイニングを用いた ニュースサイトの分類

Classification of News Sites by Text Mining

指 導 教 員　　出口 利憲 教授

岐 阜 工 業 高 等 専 門 学 校　電気情報工学科

2013E05 稲垣 天斗

平 成 ３ ０ 年（ ２ ０ １ ８ 年 ）　　２ 月 １ ６ 日 提 出

Abstract

In this research, the similarity of some news sites in Japan is calculated using text mining. Based on the similarity of each site, what kind of information is on which site is studied. We used a programming language called "R" is used to calculate the similarity. Hierarchical cluster analysis was carried out from cosine similarity, and the similarity of the articles of each news site was made into dendrogram and summarized. Since proper nouns such as person's names or organization names appear frequently in news, They were made not to divide on "MeCab" using user's dictionary. In order to check the effect of the dictionary, the result without applying the user dictionary was also calculated. To obtain the conceptual distance between words, A conceptual dictionary called "Nihongo WordNet" was used. Words which are not registered in the dictionary are changed by changing the notation and changing the language. Results were classified correctly except for two sites. It seems that the cause may be in the method in clustering and the number of word cluster.

目 次

第1章　序論

1.1　序論

昨今のデジタル機器の発達は目覚ましく、今日、我が国ではもはやパソコンやスマートフォンのようなデジタル機器を持っていない人間を見る機会の方が少なくなった。また、ついこの間までデジタル機器とは無縁であったはずの国にすら普及し始めることも珍しくない。さて、誰でもデジタル機器を用いてインターネットを閲覧する事ができる現在において、情報の取捨選択が重要になってきている。インターネットによって多くの情報を今までよりも手軽に入手できるようになったぶん、どの情報を信用すればいいのか迷っている人間が生まれてしまった。

本研究では、ネット上に数多く存在するニュースサイトの記事にテキストマイニングを使用する事で、サイトごとの特性を割り出した。特性の判定には cos 類似度を用いた。詳しい内容までは自分の目で確かめる必要こそあるものの、これによりニュースサイト間の関連度がわかる。これにより、普段使用しているニュースサイトと似た傾向のサイトを閲覧して事象に対する理解を深めたり、あえて全く異なる傾向のサイトを閲覧することが、自身とは異なった価値観に触れ、より多様な考えを身につけることができると考えられる。

第2章　テキストマイニング

2.1　テキストマイニング

2.1.1　データマイニング

データマイニングというのは、データベース内に存在する大量のデータから、有用な知識を発見する際に用いられる技術のことである。ここでいう知識というのは、データの中に見られるルール、法則のことである。有益な情報の発見というのは、少量のデータであれば人間にも可能である。しかし、顧客アンケートや売上データのような膨大な量のデータになると、人間が新しい有益な情報の発見をするのは非常に困難である。このため、コンピュータを使用した高速な処理によるデータ上の新しい有益な情報の発見が求められる。このような場面にデータマイニングが用いられる。

2.1.2　テキストマイニング

テキストマイニングというのは、前述したデータマイニングの中でもテキストデータを対象とするものである。文章からなるデータを単語や文節で区切り、そこから単語ごとの出現頻度、出現傾向、共出現の相関などを解析することで、有用の知識を発見する。テキストマイニングは、主にアンケートで消費者の声を聞いて、商品が売れる、もしくは売れない理由、改善すべき点の把握、次に何が売れるかの予測をする為に利用されている。この研究では、ニュースサイトごとに解析を行い、サイトごとの特徴を割り出すのに用いた。

2.1.3　形態素

形態素というのは、言語学における用語の一つであり、意味を持つ表現要素の最小の単位である。すなわち、ある言語においてそれ以上分解してしまうと意味をなさなくなるようなまとまりのことである。

2.1.4　形態素解析[1)]

形態素解析というのは、テキストデータを前述した形態素に分解することである。 具体的には、文書を品詞単位に分けることで、それぞれの単語の頻度を計算に使用することである。 形態素解析を行うためのツールには kuromoji、KyTea、janome など、幾つ

か種類があるが、この研究では、MeCab というソフトを使用した。

2.1.5 MeCab

MeCab はオープンソースの形態素解析エンジンである。日本語にも対応しており、単語の区切りがわかりにくい文を形態素解析することも可能である。例えば、MeCab を使用して以下のような文を形態素解析した結果を示す。

すももももももものうち

すもも 名詞

も 助詞

もも 名詞

も 助詞

もも 名詞

の 助詞

うち 名詞

このように、同じ文字が羅列しており、人間の目でも区切りが分かりづらいような文章であっても形態素解析する事が可能である。MeCab は R 言語では RMeCab というパッケージを使うことにより、R 言語上で利用することができる。

2.2 自然言語

2.2.1 自然言語

自然言語というのは、人間によって日常の意思疎通のために用いられる、文化的背景を持って自然に発展してきた言語である。また、コンピュータ言語以外の言語という意味でも使われる。両者の違いは、自然言語が人間同士で意思の疎通をするために作られてきた言語であり、コンピュータ言語は人間がコンピュータに処理をさせるために作られた言語である。自然言語においては、伝える際に曖昧な表現を用いても、相手が解釈することで正しく伝わる。しかし、コンピュータ言語では曖昧な表現は認められず、ある命令に対する動作は決まっている。なぜなら、ある命令に対する動作が複数あった場合、実行するたびに処理が変わったり、実行するごとにいちいち処理の種類を決定する必要があるからである。

2.2.2　自然言語の曖昧さ

前述した自然言語の曖昧さには二つの種類がある。一つは多義性であり、これはある単語が複数の意味を持っており解釈が複数になること。すなわち単語の意味が完全に一位に定義できないことを指す。例えば、「適当」という単語がある。「適当な量入れる」とあればこの「適当」は「ふさわしい」という意味になるが、「適当な人」とあれば、「いい加減」という意味になる。もう一つは類義性であり、これは異なる単語が同じ意味を示す場合のことである。「雷」という単語と「稲妻」という単語は同じ意味を持つ。 このような曖昧さは、コンピュータにおける自然言語の処理を難しくしている。

第3章　実験に使用した技術

3.1　実験に使用した技術

3.1.1　TfIdf[1)]

TfIdf というのは、文書中の単語の重みである。Tf は Term Frequency、すなわち文書中の単語の頻度を表す。Tf では、より多く出現する単語ほど重要であると言える。また、Idf は Inverse Document Frequency、すなわち単語の情報量を表す 。具体的には、それぞれの単語が幾つの文書内で共通して使用されているかを表す。Idf では、いくつもの文書において使用されている単語は、それほど重要度が高くないと言える。文書中の単語の重みは、この Tf と Idf の積で求める事ができる。

文書の数が N 個のとき、各文書を d_i $(i = 1, 2, \ldots, N)$ とする。また、文書における単語が M 種類のとき、各単語を t_{ij} $(j = 1, 2, \ldots, M)$ とする。 文書 d_i で単語 t_{ij} の Tf、Idf、TfIdf は次のような式で表す事ができる。

$$Tf_{ij} = \text{文書 } d_i \text{における単語 } t_{ij} \text{の出現回数} \tag{3.1}$$

$$Idf_j = -\log \frac{\text{全文書数 } N}{\text{文書に単語 } t_{ij} \text{を含む文書数}} \tag{3.2}$$

$$TfIdf_{ij} = Tf_{ij} \times Idf_j \tag{3.3}$$

3.1.2　cos 類似度

cos 類似度とは、ベクトル空間モデルにおいて、文書同士を比較する際に用いられる類似度計算法である。cos 類似度は、ベクトル同士の角度でそのままベクトル間の類似度を表すことができる。このため、三角関数で用いるコサインの通り、1 に近ければ類似しており、0 に近ければ類似していないということになる。

ベクトル $\vec{a}$ とベクトル $\vec{b}$ の cos 類似度は次のような式で求められる。

$$\cos(\vec{a}, \vec{b}) = \frac{\vec{a} \cdot \vec{b}}{|\vec{a}||\vec{b}|} \tag{3.4}$$

3.1.3　主成分分析[2)3)]

実験及び調査において、項目数が少ない時はグラフや統計量を用いてその特性を簡単に知ることが出来るが、項目数が多い時にはデータの関係が複雑になり、人間の目では

結果の分析が困難になる事が多い。このような時に、主成分分析が用いられる。

主成分分析というのは、多次元のデータを圧縮する方法である。例えば、100 次元空間のデータを 10 次元で表したいと言った時に用いられる。主成分分析を行うことにより、人間の目による結果の分析難易度が下がるだけでなく、少ない次元で表すことによって、保存するデータも少なくて済む。

方法については、以下に具体的な式を用いて説明する。P 個のデータ $x_p(p = 1, 2, \ldots, P)$ がある時、 $N(N \leq P)$ 個の主成分 $z_n(n = 1, 2, \ldots, N)$ とこれらの関係は、次式のように互いに独立な線形結合として表される。

$$z_n = \sum_{p=1}^{P} a_{pn} x_p \tag{3.5}$$

ここで z_n は第 n 主成分と呼ばれ、その結合係数 a_{pn} は以下の式を満たす必要がある。

$$\sum_{p=1}^{P} a_{pn}^2 = 1 \tag{3.6}$$

主成分が多くの情報を持つようにするためには、この結合係数を上手く決めてやる必要がある。結合係数を決める際には、データの分散に着目する。例えば、解析するデータが Figure 3.1 のような身長と体重のグラフで表す事ができるものとする。図において、x1 と x2 が共に動く軸、すなわち、テータの分散が最も大きくなる方向に着目すると、z1 という軸が出来ることが分かる。これが第 1 主成分となり、このような軸が出来るように式 3.5 の結合係数を決定するのである。しかし、この軸だけでは説明しきれない情報が存在する。このデータの場合、第一主成分で説明する事ができるのは体の大きさのみであり、身長が高いのに軽い痩せ型、もしくは身長が低いのに重い肥満体型の人間についての情報を説明することはできない。そこで次にデータのばらつきが大きい軸、すなわち第２主成分 z2 をとる、これにより、情報量の損失を最小にしながら、 データの持つ特性を把握することが出来る。

この例は２次元という、人間の目で見てもある程度は情報の把握ができる例であったがために、主成分分析の恩恵を感じづらいかもしれない、しかし、高次元のデータに主成分分析を行うことによって、得られる恩恵はわかりやすくなる。

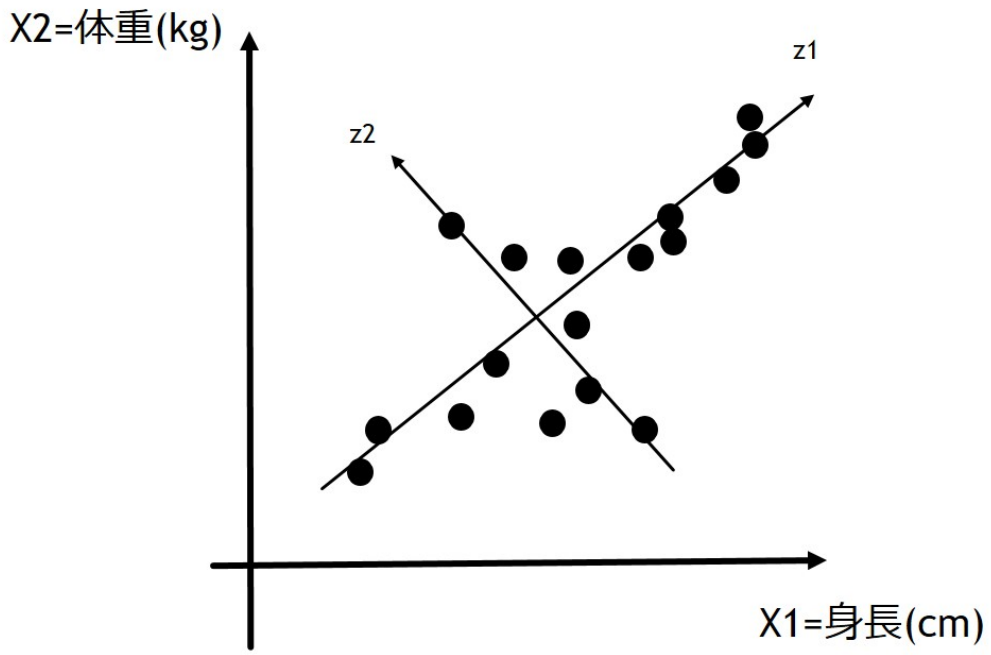

Figure 3.1　graph of height and weight

3.1.4　主成分選択

主成分分析において、主成分の数を決めることは重要な問題である。もし少なすぎると、多くの情報が失われてしまうこととなる。かと言って、多すぎると次元が減らず、主成分分析を行う意味がなくなってしまう。

主成分の選択方法としては以下のようなものがある。

- 固有値が 1 を越える主成分を採用する。
- ある固有値とその次の固有値の差が小さくなるまでの主成分を採用する。
- 累積寄与率がある値に達するまでの主成分を採用する。

固有値が 1 を超えるということは、平均と分散を共に 1 としたことで、分散（固有値）がこの標準化された値である 1 よりも大きければ、説明力のある主成分として用い得るという考えに基づいている。また、ある固有値とその次の固有値の差が小さければ、主成分の採用・非採用の区別にあまり意味はないという考えに基づいている。累積寄与率がある値に達するまでの主成分を採用するというのはデータから得られる全情報の何割かを含んでいれば良いという考えに基づくもので、普通 60〜80 パーセントに達するまでの主成分数を採用する。

累積寄与率というのは、主成分分析において、寄与率を大きい順に順次足していったものである。この寄与率というのは、あるデータ全体の変化に対して、その構成要素で

あるここのデータの変化がどのように貢献しているかを示す指標の一つである。寄与率は次式で表される。

$$P_n = \frac{\lambda_n}{\sum_{p=1}^{P} \lambda_p} \tag{3.7}$$

ここで、λ_n は n 番目の主成分の固有値を示す。このように、ある主成分の固有値が表す情報が、全ての情報の中でどの程度の割合を占めているかを表すのが寄与率である。前述したように、累積寄与率は寄与率の挿話であるため、次のように表される。

$$C_n = \sum_{i=1}^{n} P_i \tag{3.8}$$

3.1.5 LSA

LSA は latent semantic analysis の略称であり、潜在的意味解析のことを指す。潜在的意味インデキシングともいう。これは、情報検索の際に用いられる次元圧縮手法である。例えば、「車で行く」という文章と「自動車で行く」という文章があったとする。当然、「車」と「自動車」は同じ意味であり、必然的にこの二つの文章も同じ意味になるのだが、「車」と「自動車」では単語そのものが違うため、この二文は違う意味を持った文として扱われてしまうため、「車」で検索しても、「自動車で行く」という文が出てこない。このような問題を解決するために、LSA は用いられる。

3.1.6 クラスター分析[3)4)]

クラスター分析というのは、異なる性質のものが混ざり合っている集団において、互いに似たものを集めて集落（クラスター）を作成し、集団内のデータを分類するという方法のことである。クラスター分析を用いると、集団を客観的な基準に従って科学的に分類できるようになる。

クラスター分析には、大きく分けて二種類の方法が存在する。一つは分類が階層的になる階層的クラスター分析。もう一つは、あらかじめクラスター数を指定して分類する非階層的クラスター分析がある。この研究では階層的クラスター分析を利用する。階層的クラスター分析では、データ間の類似度に基づいて、最も似ているデータから順次集めていき、クラスターを形成していく。階層的クラスター分析を行うと、デンドログラム（樹形図）が出力される。デンドログラムにおいては、末端の方で結合するほど近い

関係にあると言える。階層的クラスター分析においては、集団を単にいくつかのクラスターに分類するだけでなく、クラスターが結合される過程がデンドログラムによって直感的に把握することができる。しかし、分類する対象が非常に多い場合、計算量が非常に多くなってしまい実行不可能になってしまったり、結果が不安定になったりすることがある。

具体的な結果の見方については、Figure 3.2 を用いて説明する。

A、B、C を例にして見ていくと、A と B から出た線がまず結合されている。これは A と B がこれ以降一つのクラスターとして結合されたことを表している。次に、この A、B から成るクラスターと C が結合される。これは、先ほどできた A、B から成るクラスターの中に、C が追加されたことを表している。

デンドログラムにおいては、図における下の方で結合すればするほど近い関係にあると言える。よって、図においては、A と B は非常に近く、C はそれについで近いということを表している。また、図において赤線で区切られた A〜G から成るクラスターと、H、I から成るクラスターが最後に結合しているため、これらは最も遠い関係にあるクラスターであると言える。

また、図における青線の高さを変えることで、クラスターの分割数を変更することができる。この図では、分割数は 4 である。内訳としては、A と B と C から成るクラスター。D と E から成るクラスター、F と G から成るクラスター、H と I から成るクラスターの四つである。

階層的クラスター分析において、クラスター間の距離を決める方法には、最近隣法、最遠隣法、群平均法、ウォード法といったようにいくつかの種類があるが、この研究ではウォード法を使用する。

3.1.7　ウォード法

ウォード法というのは、２つのクラスターを融合した際に、クラスター内の分散と他クラスター間の分散の比を最大化する、すなわち、クラスター内における分散がもっとも小さくなる基準でクラスターを形成していく方法である。このようなやり方のため、最小分散法とも呼ばれている。

ウォード法は計算量こそ多くなるものの、分類感度が高く、クラスター分析の手法の中ではもっとも明確なクラスターが出やすい手法であるため、代表的な手法としてあら

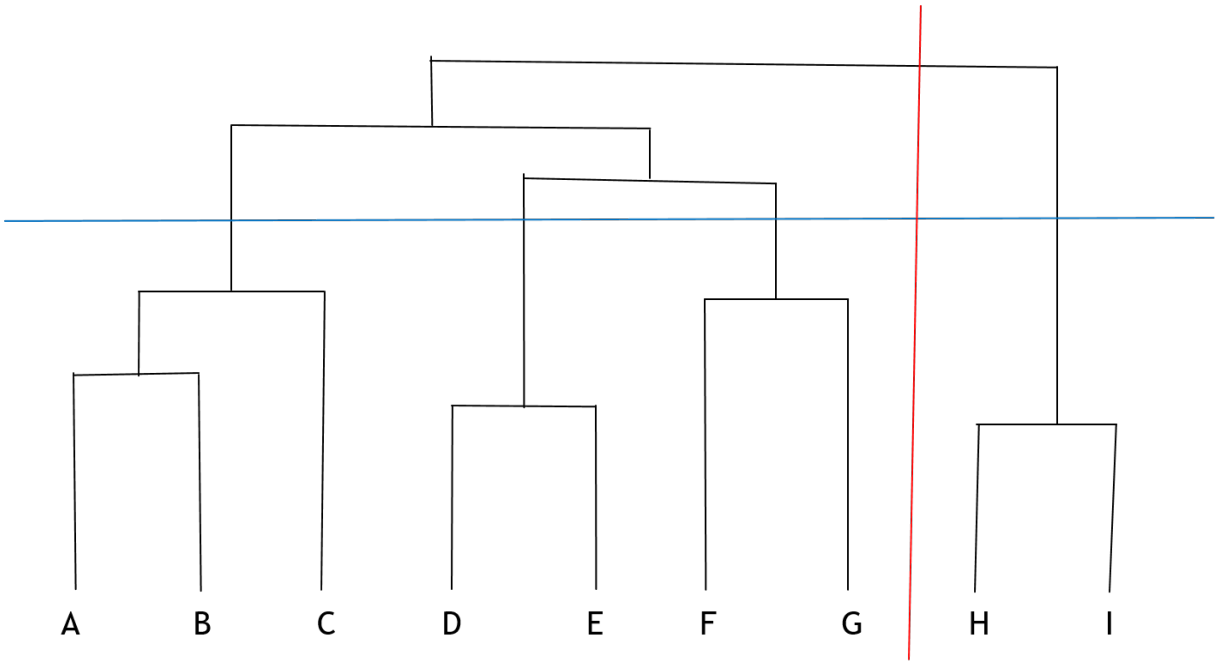

Figure 3.2　Dendrogram

ゆる場面で使用されている。以下に Figure 3.3 を用いて説明する。ウォード法においては、クラスター内での分散が最小になれば良い。各場合における分散の計算結果を左上に示す。分散が最小になったのは、A と B、C と D と E が結合した場合のため、最終的な分類結果は Figure 3.3 のようになる。

3.2　R 言語

R 言語は、統計解析向けのプログラミング言語及びその開発実行環境であり、オープンソースかつフリーのソフトウェアである。常に多くの関数が用意されているため、複雑な計算をわずか数行で実行できるという特徴がある。R 言語は S 言語を参考としてニュージーランドのオークランド大学の Ross Ihaka と Robert Clifford Gentleman により作成された。S 言語は行列を扱うことができるので、R 言語も行列を扱うことができる。R 言語のソースコードは主に C 言語、FORTRAN、そして R によって開発されている。R 言語は、ベクトル処理と呼ばれる実行機構により、柔軟な処理を簡便な記法で実現している。ここで言う「ベクトル」とは、数学的用語のベクトルとはやや異なり構造を持ったデータ集合という「リスト」に近い意味を持つ。このため、実数や複素数からなる数学上のベクトルや行列はもちろん、配列・リスト・テーブル（データフレーム）・集合・時系列などといった複雑な構造を持ったデータも、C 言語のように、int や float などの型を宣言することなく変数に代入することができる。 また、ベクトルの要素がさらにテー

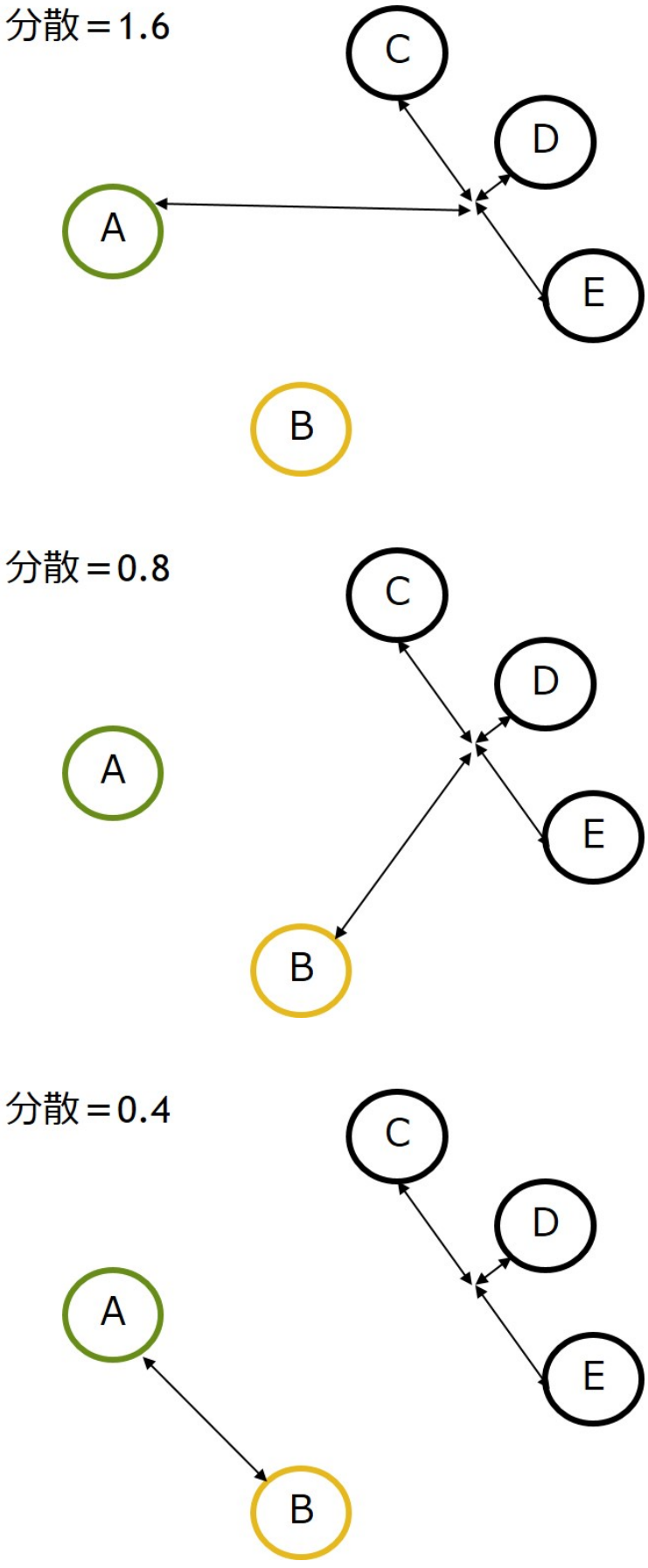

Figure 3.3　Ward's method

ブルや時系列の配列などであるといった入れ子構造であったとしても、問題なく実行できる。

3.2.1 RMeCab

RMeCab は R 言語のパッケージの一種であり、R 言語上で MeCab を利用できるようにするものである。

3.2.2 主成分分析

R 言語においては、prcomp 関数を用いることによって、主成分分析を行うことができる。また、主成分分析を行なった結果に対して、 summary 関数を用いることによって、標準偏差、寄与率、累積寄与率を一度に求めることができる。

3.2.3 LSA

R 言語においては、svd 関数を用いることによって、左特異（ターム）ベクトル、特異値、右特異（文書）ベクトルが得られる。得られた左特異ベクトルの k 列目までの転置行列と元の文書行列をかけることで、k 次元までの圧縮が可能である。

3.2.4 クラスター分析

R 言語においては、hclust 関数を用いることによって、クラスター分析を行うことができる。hclust 関数において、引数を渡すことでクラスター分析の方法を変更することができる。デフォルトは最遠隣法になっている。

3.2.5 日本語 WordNet[5)]

日本語 WordNet というのは、国立研究開発法人情報通信研究機構（NICT）によって「大規模かつどなたでもご入手いただける日本語の意味辞書」を目的として開発された日本語の概念辞書である。日本語 WordNet においては、個々の概念が synset という単位にまとめられており、他の synset と意味的に結びついている。この研究では、記事に登場する単語間の概念距離を求めるために使用した。

日本語 WordNet における synset の構造は、ある synset をルート synset とすると、そこから下位の synset が木構造のように存在しており、これが概念同士の関係を表してい

る。この synset の関係から、単語間の概念距離を求める。

概念距離を求める方法は二つの単語の synset の関係によって変わってくる。

同一の synset に存在する場合

概念距離は 0 とする。

異なる synset に存在する場合

概念距離は式 3.10 を使用して求める。

異なる synset に存在、かつ複数のルートがある場合

それぞれの距離を式 3.10 を使用して求め、それらの平均を概念距離とする。

複数の synset に属する場合

それぞれ式 3.10 を用いて求め、それらの最小値を概念距離とする。

synset の関係が見つからない場合

概念距離は 1 とする。

単語 a,b が存在するとき、それぞれのルート synset からの段数を L_a 、L_b とし、二つの共通の synset の段数を C_{ab} とする。このとき、求める概念の類似度 $S_{a,b}$ は次の式で表せる。[6)]

$$S_{a,b} = \frac{2C_{ab}}{L_a + L_b} \tag{3.9}$$

式 3.9 は最大値が 1 になるよう正規化されているので、式 3.10 で類似度 $S_{a,b}$ は概念距離 $D_{a,b}$ に変換できる。

$$D_{a,b} = 1 - S_{a,b} \tag{3.10}$$

第4章　実験

4.1　実験内容

9つのニュースサイトについて、サイト間の類似度を求めた。サイト名は以下の通りである。

- 朝日新聞
- 毎日新聞
- 産経新聞
- 神戸新聞
- 東京新聞
- 佐賀新聞
- 琉球新報
- プレシデントオンライン
- しんぶん赤旗

記事の内容は、各サイトにおいて、「所信表明演説」と検索した際にヒットした記事の中で、掲載日が2017年11月17日〜22日の記事を使用した。政治関連のニュースには、個人名、団体名が頻出するため、MeCabのユーザ辞書を適用した場合としなかった場合、日本語WordNetによる概念の置き換えを行なった場合と行わなかった場合についてクラスター分析を行い、デンドログラムを作成した。クラスター分析にはウォード法を用いた。

4.2　実験準備

類似度を用いて各ニュースサイトをクラスター分析する前に、以下の操作を行った。

- ニュースサイト内の条件を満たす記事をテキストファイルにまとめる
- R、MeCab、日本語WordNetのインストール
- R言語のパッケージ、RMeCab、proxyのインストール
- 主成分数の決定
- ユーザ辞書の作成

4.2.1　ニュースサイト内の記事をテキストファイルにまとめる

ニュースサイトごとにテキストファイルを作り、前述の条件を満たすような記事をコピーしてテキストファイルにペーストしていった。ただし、演説の原文はどのサイトにも全く同じことが書いてあるため、データには含めないものとした。また、そのままだと R で読み込んだ際に文字化けを起こしてしまうため、文字コードを UTF-8 にした。

4.2.2　主成分数の決定

R において、まず prcomp 関数を用いてテキストファイルにまとめたニュースサイトの記事に主成分分析を行い、得られた結果をもとに summary 関数を使用することによって、累積寄与率を求めた。分析の結果、累積寄与率は第二主成分で 72.95 %であったが、第三主成分で 83.34 %となり、80 %を超えた。よって、主成分数は 3 とした。

4.2.3　ユーザ辞書の作成

人名、団体名については、意図しない部分で分割され、結果に影響が及ぶのを防ぐため、ユーザー辞書によって次のように指定し、一つの単語として扱った。

安倍晋三,,,0, 名詞, 一般,*,*,*,*, 安倍晋三, アベシンゾウ, アベシンゾウ

希望の党,,,0, 名詞, 一般,*,*,*,*, 希望の党, キボウノトウ, キボウノトウ

作成方法は、まず、上記のような内容が書かれている csv ファイルを作成し、ターミナル上でコンパイルした。その後、MeCab の設定ファイルに作成した辞書のパスを指定した。また、ユーザ辞書の効力を示すため、これ以降の操作はユーザ辞書を適用した場合、適用しなかった場合の両方について実験を行なった。

ユーザ辞書に登録した単語は、次の通りである

- 安倍晋三
- 二階俊博
- 石破茂
- 細野豪志
- 枝野幸男
- 玉木雄一郎
- 大塚耕平
- 志位和夫

- 習近平
- 総理大臣
- 希望の党
- 本会議
- 森友学園
- 加計学園
- 施政表明演説
- 所信表明演説
- 郵政解散
- 日本維新の会
- 日本共産党
- 私物化
- 少子高齢化
- 小泉純一郎
- 獣医学部
- 文部科学省
- 文科委員会
- 政府予算案
- 日米首脳会談
- 加計孝太郎
- 小池百合子
- 立憲民主党
- 保守色
- 米中首脳会議
- 長島昭久
- 山口那津男
- 空気清浄機
- 認可保育施設
- 東アジア地域包括的経済連携
- 筆頭副幹事長

- 憲法尊重擁護義務

4.3 cos 類似度による類似度計算

TfIdf を重要度として cos 類似度を計算し、ニュースサイト間の類似度を求めた。

手順は以下のようにして行なった。

1. TfIdf の計算
2. 形態素解析した名詞を csv ファイルに書き出し
3. 日本語 WordNet での概念距離計算
4. 概念距離の書き出し
5. R での概念距離読み込み
6. cos 類似度の計算
7. 類似度の計算結果をもとにクラスター分析
8. 分析結果をデンドログラムにまとめる。

4.3.1 TfIdf の計算

パッケージマネージャーから RMeCab パッケージをロードし、docMatrix 関数を用いて、引数 pos に”名詞”、weight に”tf”を指定して Tf を求めた。Idf については、式を R 上で計算し、これと先ほどの Tf の結果をかけることによって、TfIdf とした。

4.3.2 解析結果の書き出し

write.table 関数を用いて、文書内に登場した名詞を csv ファイルに書き出した。名詞については、文書行列の行の名前を rownames 関数を用いて抜き出すことによって得た。

4.3.3 日本語 WordNet を用いた単語間概念距離計算

日本語 WordNet の synset を利用し、記事に登場する名詞間の概念距離を計算した。この時、日本語 WordNet 内に登録されていなかった名詞が存在した。ここで、日本語には、平仮名、片仮名、漢字の三種類の文字があることに着目した。例えば、「あいさつ」という単語がある。この単語をそのまま入力しても、日本語 WordNet で概念距離を求めることはできない。ここで、「あいさつ」という単語の書き方を変えてみる。まず、片仮名で「アイサツ」と入力してみる。この場合は、「あいさつ」と入力した時と変わらなかった。

しかし、漢字で「挨拶」と入力したところ、次のような結果が得られた。

挨拶 06632358-n salute

このように、文章内で平仮名で書かれている単語は、漢字に置き換えると概念距離の計算が可能になる場合がある。逆にいうと、日本語 WordNet は平仮名、片仮名、漢字による表記を全て別のものとして扱っているため、日本語 WordNet で日本語の文章に登場した単語の概念距離を計算する際には、このような表記に注意する必要がある。また、日本語 WordNet は日本語の概念辞書であるが、英語も登録されている。このため、日本語では登録されていなくても、英語であれば登録されている単語が存在していることに着目した。例えば、「1 月」という単語がある。この単語は、平仮名のまま日本語 WordNet 内に登録されているかどうかを調べても、日本語 WordNet に登録されていないため、概念距離を求めることができない。しかし、これを「January」とし、日本語 WordNet 内に登録されているかどうかを調べると、

january 15210045-n jan

という結果が返ってきた。このことから、日本語 WordNet において、「1 月」という単語の概念距離を計算したい場合は、「January」という単語に置き換えれば計算可能であるということになる。さらに、日本語には、2.2.2 項で述べた「適当」のように、一つの単語が複数の意味を持っていることが多い。先ほど日本語 WordNet に「挨拶」という単語を入力した結果を示したが、実際のところは、下のようになっていた。

挨拶 06632358-n salute

挨拶 06630459-n well-wishing

挨拶 06630017-n salutation

ここで、各単語の意味は次 [7] のようになっている。

1. salute 敬礼する、挨拶する、讃える、賞賛する
2. well-wishing 成功の好意を広げている、ある人から他の人への好意の表現
3. salutation 挨拶、挨拶の言葉、手紙の書き出しに使う挨拶の文句

置き換えの方法であるが、「置き換える対象の単語 Tab 置き換え後の単語」をまとめたテキストファイルを作成した。具体的には、次のようなものが並んでいる。

独自 original

重視 emphasis

明記 specific

置き換えを行なった単語は、Table4.1 である。

また、今回のデータに含まれる単語の中には、英語で置き換えないと概念距離が計算できず、なおかつ意味が複数個存在する単語もあった。このような単語に関しては、英訳する前から直接ある概念に置き換えた。具体的には、次のようなものが並んでいる。

中国 03018209-n

苦言 07208930-n

悲願 05791602-n

右側の数列が、概念を表している。置き換えを行なった単語は Table4.2 の通りである。

上記二つは、それぞれ別のテキストファイルにまとめた。なお、上記二つの方法を持ってしても日本語 WordNet 内に発見されなかった単語については、そのままにしてある。このような単語の置き換えを行う理由は、英語にはあって日本語にはない意味を除外するためである。

4.3.4 概念距離の書き出し

日本語 WordNet により求めた概念距離を csv ファイルに書き出した。この時、結果の比較を行うために、上記の置き換えを行なった場合と行わなかった場合の両方について、csv ファイルの書き出しを行なった。

4.3.5 R での概念距離の読み込み

上記の csv ファイルを R 上で読み込んだ。

4.3.6 cos 類似度計算

パッケージマネージャーから proxy パッケージをロードし、dist 関数を用いて R 言語上で cos 類似度を求めた。データには TfIdf の行列を指定し、cos 類似度の行列を計算した。

4.3.7 クラスター分析

R 言語の hclust 関数を用いて TfIdf の cos 類似度の行列を入力し、引数 met には”ward.D2”を指定し、クラスター分析を行なった。この「ward.D2」は、ウォード法を表している。

Table 4.1　Word replacing list

日本語	英語
安保	security
あいさつ	salutation
独自	original
重視	emphasis
明記	specific
１２月	december
１１月	november
１月	january
７月	july
５月	may
謙虚	humility
９月	september
拉致	abduction
急速	rapid
集約	integration
６月	june
大改革	reform
施政	government
無料	free
無償	free
１０月	october
８月	august

日本語	英語
新規	new
罰則	penalty
年収	income
帽子	hat
もろさ	brittle
共産	communism
貧しい	poor
二つ	two
当選	chosen
はじめ	beginning
五輪	olympics
デビュー	debut
保育園	kindergarten
重視	emphasis
カギ	key
穏健	moderate
羅列	enumation
積極	positive
にくい	hate
可能	possible
特別	special
暴走	runaway

Table 4.2　Concept replacing list

日本語	概念	概念 ID
イノベーション	invention	03582658-n
中国	china	03018209-n
苦言	complaint	07208930-n
悲願	wish	05791602-n
先送り	deferment	01066881-n
アピール	appeal	07186828-n
介護	care	00654885-n
理念	idea	05833840-n
使命	mission	00730984-n
試算	admission	05802185-n
進学	humility	07215948-n
北方領土	territory	05999134-n
待機	wait	01063939-n
過激	radical	14874196-n
被害	damage	07339653-n

4.3.8　デンドログラム作成

proxy パッケージの plot 関数を用いることによってデンドログラムを導き出した。この時、入力は cos 類似度の行列とした。また、画像の出力には postscript 関数を用いた。

4.4　実験結果

実験結果を Figure 4.1 から Figure 4.4 に示す。それぞれにおけるユーザ辞書、日本語 WordNet での単語の置き換えの有無は図タイトルの通りである。

4.4.1　実験結果の評価方法

今回の実験において、結果の評価は Figure 4.1 のユーザ辞書の適用も日本語 WordNet による単語の置き換えも行なっていない場合のデンドログラムと、Figure 4.4 の両方と

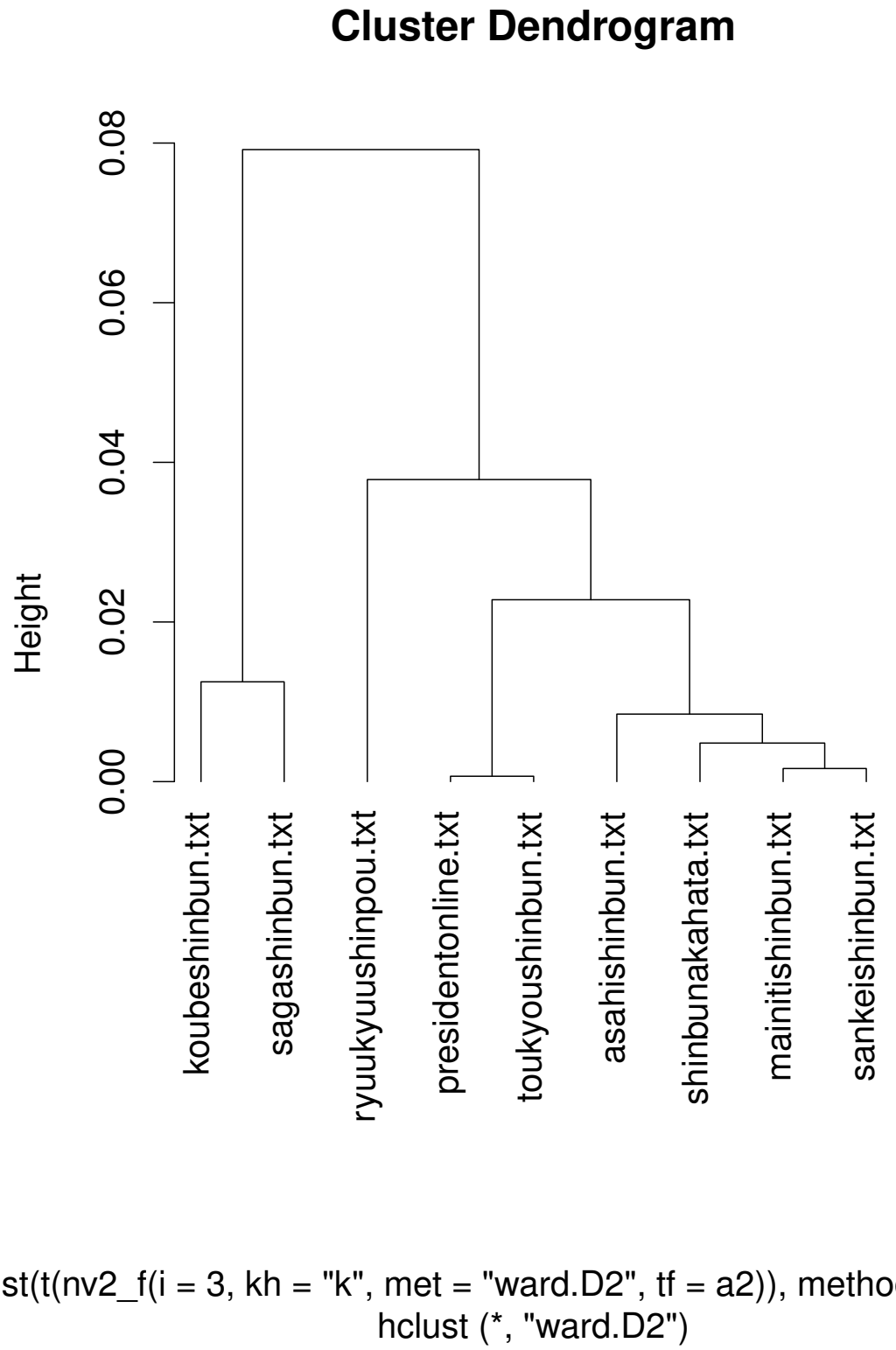

Figure 4.1 without user's dictionary and word replacing

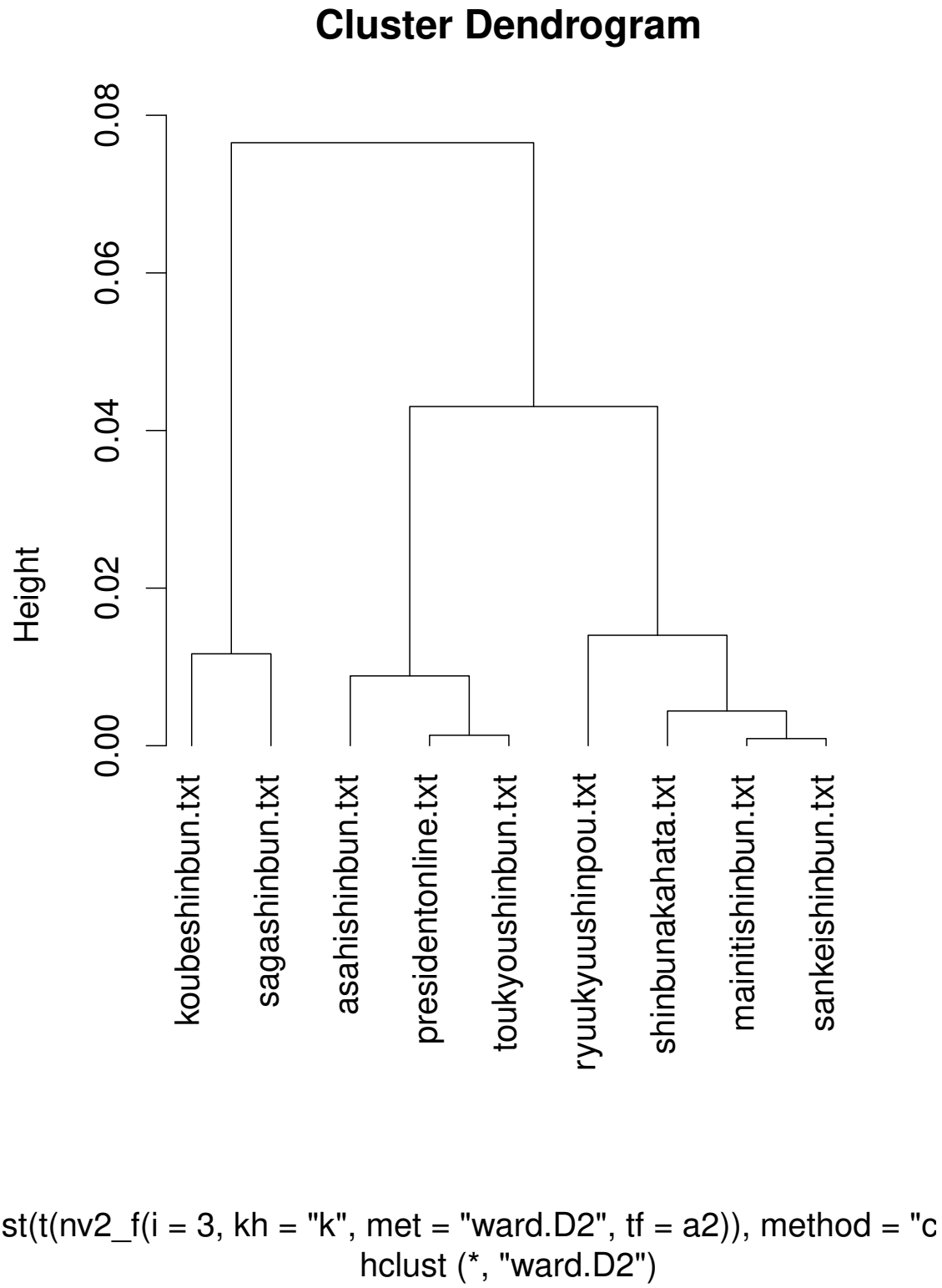

Figure 4.2 with user's dictionary without word replacing

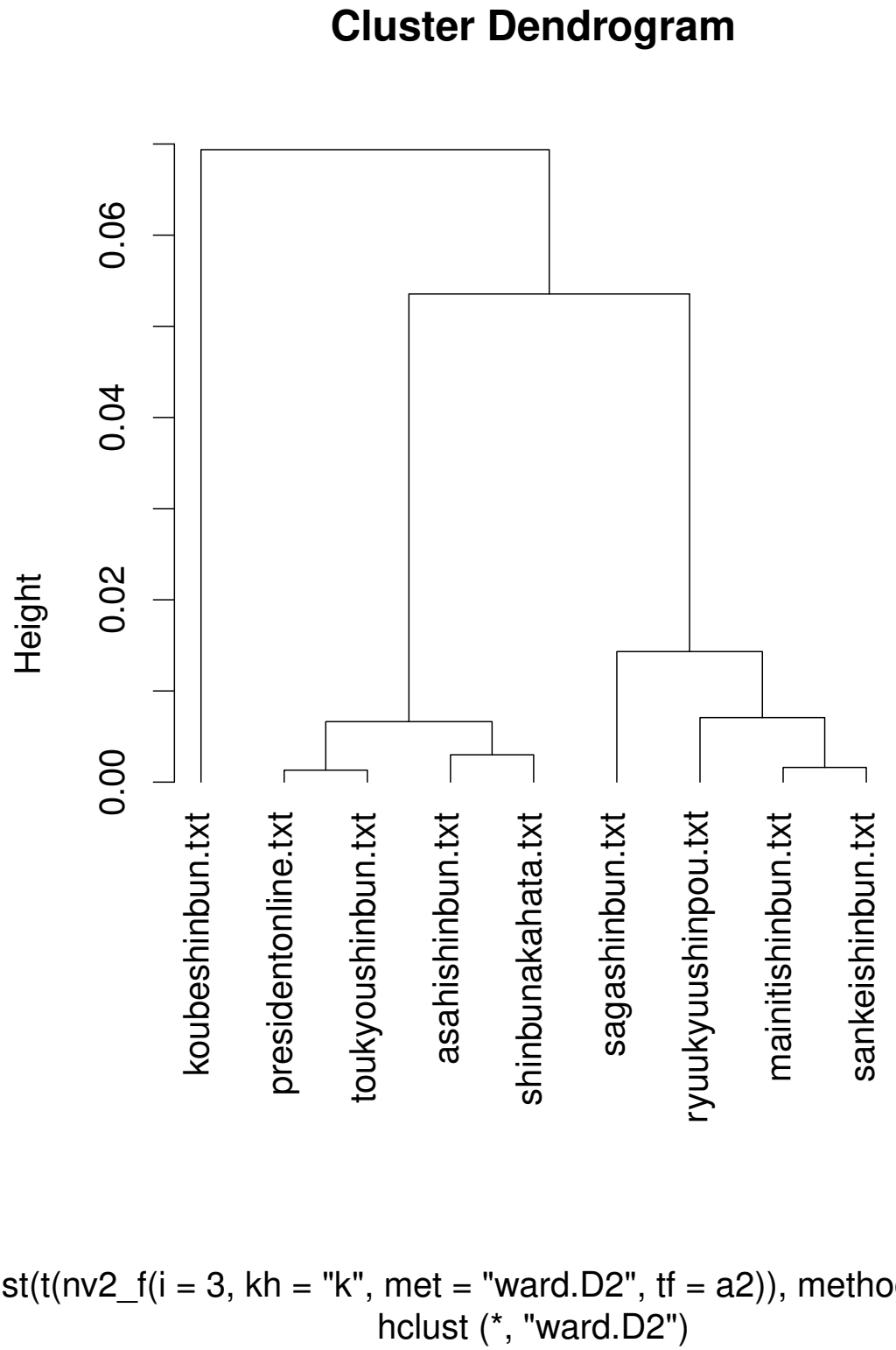

Figure 4.3 without user's dictionary with word replacing

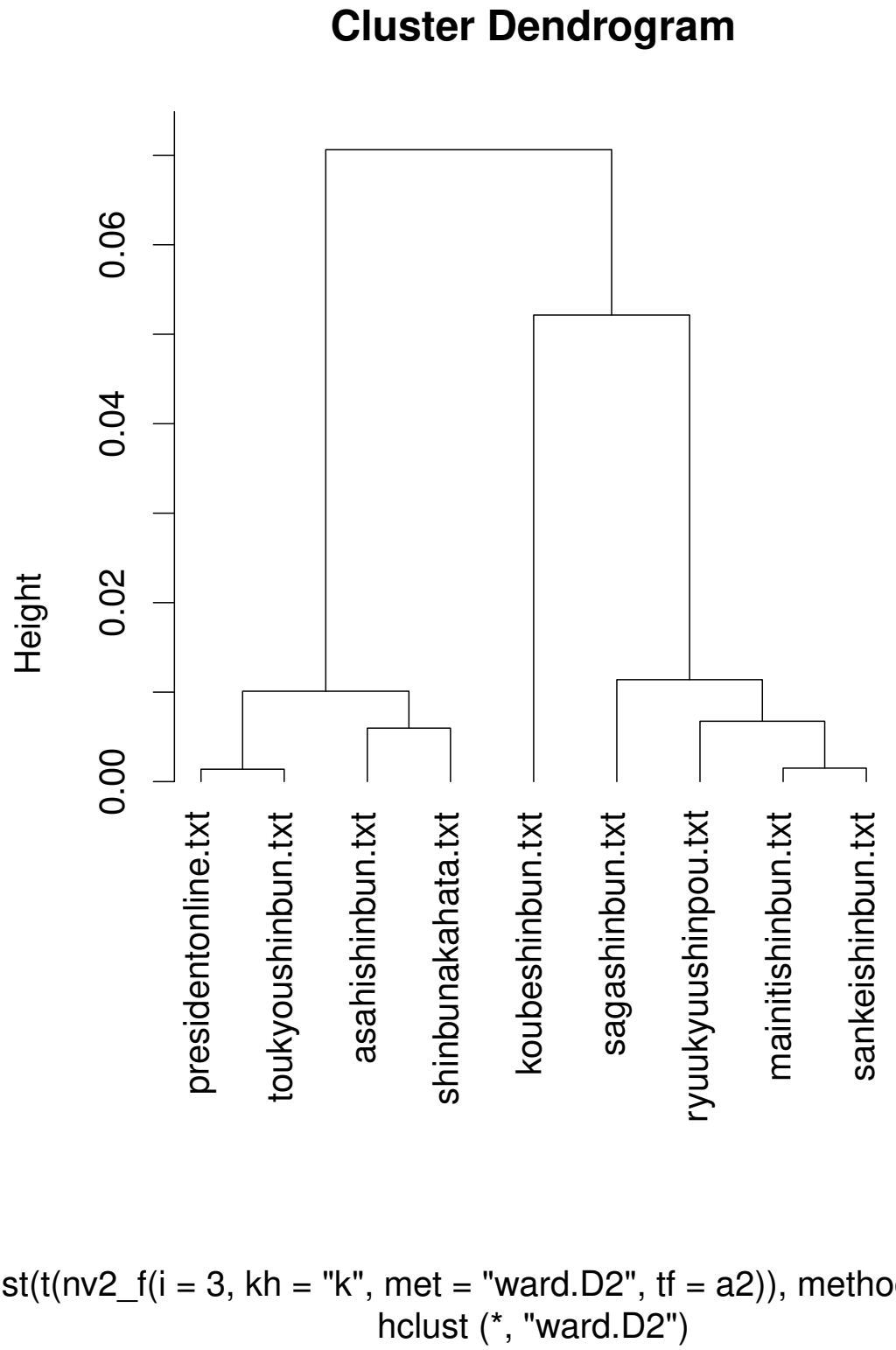

Figure 4.4 with user's dictionary and word replacing

も行なった場合とを基準として考察を行うことにした。また、デンドログラムのみではクラスタリングの正確さを測ることは不可能であるため、記事の見出し、および内容も考慮してどのような条件であればより正確にクラスタリングを行うことが可能であるかを考察した。

4.4.2 結果の比較

ユーザ辞書を適用した場合

Figure 4.1 と Figure 4.2 を比較する。これらの違いは、ユーザ辞書を適用してあるか否かである。

両方とも、神戸新聞と佐賀新聞から成るクラスターとそれ以外から成るクラスターとに分かれている。それ以外から成るクラスターについて見てみると、プレシデントオンラインと東京新聞、そして、毎日新聞と産経新聞、これら二つとしんぶん赤旗が共通のタイミングで結合していることがわかる。この他二つ、琉球新報と朝日新聞の結合するタイミングが異なっていた。

Figure 4.3 と Figure 4.4 を比較する。

どちらも、大きく分けて三つのクラスターが存在していることがわかる。一つ目は神戸新聞、二つ目はプレシデントオンライン、東京新聞と朝日新聞、しんぶん赤旗の二つの集まりから成るクラスター、三つ目は毎日新聞と産経新聞、これらと琉球新報、これらにさらに佐賀新聞を加えたクラスターである。両図においては、これら三つの結合のタイミングのみが異なっている。

日本語 WordNet での単語の置き換えを行なった場合

Figure 4.1 と Figure 4.3 を比較する。これらの違いは、日本語 WordNet での単語置き換えを行なったか否かである。

置き換えを行なった場合、神戸新聞のみが、他と大きく異なっているという結果になった。また、ここでも毎日新聞と産経新聞、そして、プレシデントオンラインと東京新聞は共通のタイミングで結合していた。

Figure 4.2 と Figure 4.4 を比較する。

毎日新聞と佐賀新聞から成るクラスター、プレシデントオンラインと東京新聞から成るクラスターが存在しているのは共通であったが、その他の結合の仕方が全て異なって

いた。

両方とも適用した場合

Figure 4.1 と Figure 4.4 を比較する。これらの違いは、ユーザ辞書を適用してあるかと、日本語 WordNet での単語置き換えを行なったか否かである。

ここでも毎日新聞と産経新聞、そして、プレシデントオンラインと東京新聞は共通のタイミングで結合していた。

4.4.3 まとめ

どのデンドログラムにも共通して登場したのが、毎日新聞と産経新聞から成るクラスターとプレシデントオンラインと東京新聞から成るクラスターの二つである。これらはユーザ辞書、および単語の置き換えの有無に関わらず、最速で結合していた。この四社以外の結果は操作によって決まっているという印象を受けた。ユーザ辞書の適用によって産経、毎日と赤旗、琉球が、また、プレシデント、東京と朝日が結合し、日本語 WordNet による単語の置き換えによって産経、毎日と琉球、佐賀が、また、プレシデント、東京と赤旗、朝日が結合した。両方を適用した場合は、ユーザ辞書を適用した際には産経、毎日と結合していた赤旗が、両方とも適用した場合、朝日、プレシデント、東京と結合していたため、結果に与える影響は日本語 WordNet による単語の置き換えの方が強いと思われる。

4.5 考察

4.5.1 何故このような結果になったか

今回の研究においては、日本語 WordNet による単語の置き換えが結果に与える影響が大きく、ユーザ辞書の適用が結果に与える影響はあまり大きくなかった。ユーザ辞書があまり影響を与えられなかった理由については、置き換えた単語が「安倍晋三」「小池百合子」のような人名や、「日本共産党」「希望の党」のような団体名が大半を占めていたためであると考えられる。クラスター分析を行う前に日本語 WordNet による概念距離の計算を行うのだが、概念距離を求められなかった単語を見ると、「安倍」「小池」など、人名に含まれていた単語が多く見受けられた。当初の想定では、例えば「小池」であれば「小」と「池」に分割されてしまうと考えていたのだが、実際はユーザ辞書に登録したほ

ぼ全ての人名が、苗字と名前に別れるだけでそれ以上変な分かれかたはしていなかった。このため、適用していようとしていまいとこれら人名の概念距離は日本語 WordNet では概念距離を求められなかったため、結果に大きな影響を与えることができなかったのであると考えられる。

一方、日本語 WordNet における単語の置き換えは、一度概念距離を求められなかった単語を、概念距離が求められるように置き換えたものであるため、置き換え後は確実に概念距離を求めることができる。このため、置き換え前と置き換え後では、概念距離を求められる単語の数が大きく異なってくる。したがって、ユーザ辞書の適用よりも、日本語 WordNet における単語の置き換えの方が結果に大きく影響を及ぼすことができると考えられる。

4.5.2 クラスター分析は正しく行われたか

実際に記事を読んで見る。

まず、どの場合でも最初に結合していた毎日新聞と産経新聞を見比べる。

両社とも首相が所信表明演説において語ったことそれぞれの内容についてと、所信表明演説では語られなかったことについての野党からの批判について、文体や内容の順番こそ異なっていたものの、かなり似たような内容が書かれていた。

次に、プレシデントオンラインと東京新聞を見比べる。

プレシデントオンラインの記事は、主に各新聞社の社説を引用して、どの新聞社がどのような意見を持っているか、それを受けて、首相は今後どのような政策を行なっていけば良いのかが書かれている。また、唯一首相がエイブラハム・リンカーンに影響を受けているのではないかということが書かれていた。

一方、東京新聞では、社説の引用などは一切されておらず、何党の誰々がこういう意見を述べていたというような内容と、通常国会における煙害 (タバコの匂いや副流煙による被害) についての内容であった。

先の毎日新聞と産経新聞と同じく、この二社は最速で結合していたのだが、実際に類似しているとは言い難いと言える。しかし、プレシデントオンラインの中に興味深い一文があった。それは、「トランプ氏は共和党であり、左派である朝日とはスタンスは大きく違うからだ。」と言った文である。「左派」という言葉は本来、急進的、革新的、革命的な人物、政治団体のことを指すのだが、一方で社会主義的、共産主義的な団体という

意味でも用いられている。最終的な結果を見てみると、上記二組の次に朝日新聞としんぶん赤旗が結合している。しんぶん赤旗は日本共産党が運営しているニュースサイトであり、正しく分類できていると考えられる。

琉球新報は記事の分量が多く、書いてある内容は所信表明演説で語ったこと、語られなかったことを軸に首相の政治に関する態度に対する言及といったものだった。この通り内容に関しては毎日、産経と似ており、比較的早い段階で結合したのも納得できる。

神戸新聞は対象となった記事の文量の少なさに加え、意見が三文程と最小限に留められており、あとは演説でこう言った話が出た。それを受けて誰々がこう言っていたのような事実のみが淡々と書かれていた。

佐賀新聞は記事の分量こそそれなりにあったものの、「生産性革命」や「人づくり革命」のような産業面の政策に関する記事が約半数を占めていた。また、多くの新聞社が触れていた森友学園、および加計学園に関する内容が少なかった。

神戸、佐賀の二社は内容そのものが他のサイトと比べても特異であると言える。これら二社の結合のタイミングが比較的遅いことが多かったのも納得できる。

第5章　結論

5.1　結論

本研究では、9つのニュースサイトについて、テキストマイニングを使用することで、ニュースサイト間の類似度を求め、どのサイトにどのような情報な記載されているかといったような、サイトごとの特性を割り出そうと試みた。

方法としては、まずTfIdfを計算し、形態素解析をおこなった。その後、日本語WordNetにおいて、記事に登場した単語の概念距離を計算した。これらの結果からcos類似度を計算し、クラスター分析を行い、分析結果をデンドログラムにまとめた。これらの操作をユーザ辞書の作成、適用の有無、日本語WordNetにおける単語、概念の置き換えの有無を変更して行なった。

ニュースサイトの記事には、人名や団体名のような固有名詞が極めて多く含まれていると考えられたため、まずはユーザ辞書とよばれるユーザが自由に作成できるMeCabの辞書を作成した。このユーザ辞書というのは、複数の単語が含まれる複合語などを扱う際に、意図しないところで分割されてしまうのを防ぐ意味合いで使用される。これにより、記事内に登場する人物の苗字、名前や団体名が変なところで分割され、日本語wordNetが文章に全く関係のない単語の概念距離を求めてしまうのが防がれるのではないかと考えた。しかし、このユーザ辞書の作成では、結果に大きな変化を与えることはなかった。考えられる理由としては、そもそもMeCab上で、ある程度有名な苗字、及び名前に関しては、分割されることなく一つの単語として認識できること、そして、人名が分割されていようとされていまいと、日本語WordNetで単語ごとの概念距離を求める際に、人名には対応していないため、日本語WordNet側からしてみれば概念距離を求められない複数の単語が、概念距離を求められない一つの単語に集まっただけで、どちらにせよ概念距離が求められないことに変わりないからであると考えられる。ユーザ辞書は、分割されてしまう単語を分割されないようにし、結果に影響が及ぶのを防ぐ役割で用いられるため、今回のような例では、この機能を生かすのは難しいのではないかと考えられる。

概念距離が求められなかった単語については、ひらがなで書かれていたものを漢字で、日本語で書かれていたものを英語でといったように、書き方を変えれば概念距離を求められる例が存在した。このため、概念距離を求められる単語を少しでも増やすため、単語の置き換えを行った。こちらはユーザ辞書と違い、置き換え前と置き換え後で結果が

大きく異なった。これは、ユーザ辞書は適用しても概念距離を求められる単語が増えるというわけではなかったが、単語の置き換えは、一度計算に失敗した単語を計算できるようにするというものであるため、全く確認せずに適当に置き換えでもしない限り、確実に成果がでるものである。このため結果が大きく変化するのは、むしろ当然であるといっても良い。

クラスター分析そのものの結果は概ね正しく類似度を求められていた。しかし、あまり関連性がないと思われるプレシデントオンラインと東京新聞が最速で結合してしまうという事態が起きた。この現象は、ユーザ辞書の有無、日本語 WordNet における単語の置き換えの有無に関わらず起こっていた。よって、原因は今回の研究では変更していない部分にあると言える。

考えられる可能性は二つ。一つはクラスター分析の方法である。今回は最も優れている、最もよく使われているという理由でウォード法を用いてクラスター分析を行なった。しかし、クラスター分析の手法はウォード法だけというわけではない。特に今回は一般的にテキストマイニングが用いられる事例とは大きく外れた試みを行なっているため、検討の余地は大いにあると考えられる。

もう一つは主成分数である。今回は先に主成分数を決定し、そのあとに日本語 WordNet による概念距離の計算を行なったため、主成分分析に用いた単語数と実際に概念距離が求められた単語数は異なっている。このため、実験における計算の順序の順序を変更することで、より納得のいく分類ができるようになるのではないかと考えられる。

謝辞

本研究を進めるにあたり、授業や会議というご多忙の中ご指導いただいた指導教員の出口利憲先生、また、学会発表の準備中にも関わらず、研究に対するアドバイスをくださった服部修平先輩に感謝の意を表します。

参考文献

1) 石田基広, “R によるテキストマイニング入門”, 森北出版株式会社 (2008).

2) 加納学, “主成分分析”, 京都大学大学院工学研究科化学工学専攻プロセスシステム工学研究室, (http://manabukano.brilliant-future.net/document/text-PCA.pdf) (1997).

3) マクロミル https://www.macromill.com

4) 構造計画研究所 http://www.kke.co.jp

5) Francis Bond, Timothy Baldwin, Richard Fothergill and Kiyotaka Uchimoto (2012) Japanese SemCor: A Sense-tagged Corpus of Japanese in The 6th International Conference of the Global WordNet Association (GWC-2012), Matsue.

6) 川島貴広, 石川勉,“言葉の意味の類似性判別に関するシソーラスと概念ベースの性能評価”, 人工知能学会論文誌 20 巻 5 号 B (2005)

7) weblio 英和辞典・和英辞典 https://ejje.weblio.jp