NAIST-IS-MT9651055

修士論文

特徴ベクトルを用いた設計プロトコルの分析と構造化

塩崎 敏也

1998 年 2 月 13 日

奈良先端科学技術大学院大学
情報科学研究科 情報処理学専攻

本論文は奈良先端科学技術大学院大学情報科学研究科に
修士 (工学) 授与の要件として提出した修士論文である。

塩崎 敏也

指導教官： 西田 豊明 教授
松本 裕治 教授
武田 英明 助教授

特徴ベクトルを用いた設計プロトコルの分析と構造化*

塩崎 敏也

内容梗概

設計中の会話を録音したプロトコルデータのような議事録には、設計ノウハウやアイデアなどの設計知識が含まれており、それを構造化しユーザに読みやすく提供するような方法が望まれている。そこで本研究では、キーワードベクトル法を利用したシステムを提案し議事録からの議事の流れの可視化を目指した。具体的には文書の全ての行についてその前後を特定の長さだけ切り出し、それぞれのキーワードベクトルを比較することによって文書全体での話題の変化を捉える。この情報から文書の区切りを発見し文書をブロック化する。さらに得られた文書ブロックでキーワードベクトルを計算し比較することによって関係を発見し、構造化を実現する。その結果、本手法によって議事録の中に設計過程を読みとることができた。また、自転車の荷台の設計におけるプロトコルを例題に実験を行ない、本手法におけるパラメータや用いるキーワードに対する調査を行なった。

キーワード

設計プロトコル, キーワードベクトル法, 文書の構造化

*奈良先端科学技術大学院大学 情報科学研究科 情報処理学専攻 修士論文, NAIST-IS-MT9651055, 1998年2月13日.

A study of keyword-vector based document structurization method for design protocols *

Toshiya Shiozaki

Abstract

The aim of this research is to develop a system which visualizes design knowledge for designers. we propose a method based on keyword vectors to divide the protocol data recorded during the design into important blocks of information and link them. By calculating cosine value of keyword vectors of adjacent parts of text blocks,we can detect points where topics shift to another. As a result,we can obtain text blocks each of which is related to some topic. Furthermore,we can find links between those blocks by caluculating cosine values of keyword vectors between them. We made experiments against a test case of bicycle design and analyzed the effectiveness of the method.

Keywords:

design protocol,keyword-vector method,document structurization

---

*Master's Thesis, Department of Information Processing, Graduate School of Information Science, Nara Institute of Science and Technology, NAIST-IS-MT9651055, February 13, 1998.

目 次

図 目 次

表 目 次

第1章

序論

1.1 はじめに

現在、高度な生産技術の出現によって大量に物が生産されるようになって久しい。その結果、消費者一人一人はどこにでもあるような製品にはあまり興味を示さなくなり、より消費者の立場に立って生産された製品を望むようになった。消費者が持つこのような要求に応えるために、生産者は従来の製品にない新しいアイデアを含む製品を素早く開発する必要に迫られている。

このような状況において生産者は効率良く製品を設計し生産する必要がある。特に設計に着目した場合、過去の設計事例や経験を再利用することによって生産性や品質が向上するであろう。

ここで設計議事録や設計メモなどの利用が考えられる。設計議事録や設計メモには、設計におけるひらめきやアイデアなどの情報が含まれており、知的な設計を支援する上で重要な役割を果たすと考えられる。

しかし、そのような文書は一般的に全体が大きすぎる上、部分部分では、内容の薄い部分が多く、有益な情報を含むところを探すことが難しいという状況にある。

そこでこのようなドキュメントを構造化することによって、有益な知識を容易に取り出せるようにし、設計をする上でより効率的で、創造的な設計につながると考えられる過去の設計に関する知識を有効に活用できるようにすることが求められている。

1.2 研究の目的

本研究の目的は、設計議事録のような構造化されていない文書の再利用を目指し、その分析と設計者が理解しやすいような形で構造化する方法の提案である。またその提案した手法を実際の設計議事録に適用し有効性を調べることである。

1.3 研究の方針

設計議事録のような情報は、重要な情報が含まれている。しかし、後で設計者がその議事録を読んで理解しようとした場合、議事の流れを理解するのに時間と労力を費やす。

そこで本研究では、議事の流れをとらえることによって議事録の構造化を目指す。具体的には議事録を意味のある断片に分割し、それらの断片の意味上の関係を発見することによって、議事を時間的ではなく意味上の流れの形成を目指す。

双方の手法において、我々は文書の意味を捉えるための単語（あるいは熟語）の集合（キーワード）に基づく方法を提案する。特に設計に関する文書や専門文書の場合、単語の意味が文の内容に深く関係することが多く、専門用語や設計一般に関する語、物の名前などに注目することによって大きな成果が得られると考えたからである。

Salton[1] らのキーワードベクトル法ではある文章の意味をそこに出現する単語の出現頻度（キーワードベクトル）で表し文書検索の手法としている。我々はこのキーワードベクトル法を逆手にとって、分割後の隣接する文書のキーワードベクトルの変化が最も大きくなるように文書を分割する手法を提案する。

分割された文書間の関係を発見し提示する手法においても我々はキーワードベクトルに基づく方法を用いている。具体的には、全ての分割された文書間のキーワードベクトルの類似度を計算することによって行なっている。

これらの方法は、キーワードの選択に依存すると考えられる。そこで、本研究では以下のような 3 種類のキーワードを準備しそれぞれ比較検討を行なう。

- 人手により抽出したキーワード
- オントロジーにおける概念

- 出現頻度順に半自動的に抽出したキーワード

また、人手により抽出したキーワードを用いる方法において、設計者に対する一般性を調べた。

1.4 本論文の構成

第 2 章では、本研究が提案するキーワードを用いて文書の流れを捉える方法について説明する。第 3 章では、本研究が提案する議事録の分割法の結果とそのパラメータの設定に関する考察を述べる。第 4 章では、第 2 章、第 3 章で述べた理論をもとに実験を行ない分析をする。第 5 章では、複数の設計者の場合を考えて、第 2 章、第 3 章で述べた理論をもとに分析する。第 6 章では、第 2 章で述べた理論を基に第 4 章で得た結果から、次の段階の実験を行なう。第 7 章では、実験結果の考察をおこなう。第 8 章では、結論と今後の課題について述べる。

第 2 章

文書の構造化の手法

本研究では、議事録の中から議論の流れを捉えるために、基礎となる技術としてキーワードベクトル法を用いる。文章を構造化するために、

- キーワードベクトルを用いて文章を意味あるブロックに分割する。
- キーワードベクトルを利用して分割されたブロックそれぞれに対してキーワードベクトルを計算し、それらのブロック間に関係を発見する。

の順で処理していく。

2.1 キーワードベクトル

キーワードベクトルは、文書中に出現する各キーワードがベクトルの成分にあたる。そしてそれぞれの成分の値は文書中で出現するキーワード数の総和を表しているものである。

2.2 キーワードベクトルを用いた文書の分割

文書検索の一手法である Salton らのキーワードベクトル法は検索キーワード群に最も近いキーワードの出現分布（キーワードベクトルの角度が小さい）を持つ文書を捜し出す。それに対して我々の文書分割の手法は、分割後の任意の隣接する文書のキーワードベクトルの角度が最も大きくなるように文書を分割する。つまりあらかじめ与えられたキーワードについて検索が行なわれた場合に検索されるべき部分とそうでない部分（キーワードベクトルの角度の違い）が特に分割後の隣接する文書間において明確に分離するような文書の分割を求めていると言える。

2.2.1 理想的な手法

文書Dを任意の分割任意の分割 (s) によって n 個の部分文章 $d_{s1}, d_{s2}, \ldots, d_{sn}$に分割し、それらのキーワードベクトルを $w_{s1}, w_{s2}, \ldots, w_{s3}$とした場合、隣合う分割された文書のキーワードベクトルの角度の変化を評価する。たとえば、

$$\sum_i (w_{si} \cdot w_{si+1}) / (|w_{si}| * |w_{si+1}|) \tag{2.1}$$

が最小となる分割 s を発見する。ただし・はベクトルの内積を表す。

しかしこの手法をそのまま実際の文書に適応するには幾つかの問題点が存在する。

- 短く切った文章は内容が特殊化しているのでキーワードベクトルも特殊化してしまう。従って、分割の最小単位(一文)で切った場合にキーワードベクトルの変化が最大になってしまう。
- 任意の分割を計算することは文書の長さ $|D|$ に対して $O(|D|^n)$ 時間の計算時間を必要とし実用的ではない。

よって我々は以下に述べるような手法を用いて近似的に文書の分割点を求める。

2.2.2 近似法

まとまった内容を議論していると思われる文書の長さを l、文書 Dをある部分 x の前後 lの部分を d_{xl-}, d_{xl+}とすると、その前後 2 つの部分のキーワードベクトルを比較することによって近似的に x において文書 Dの内容がどの程度変化したかを求めることができる。ここで文書 Dの x における内容の変化を次のようにおく。

$$c_x = \frac{w_{xl-} \cdot w_{xl+}}{|w_{xl-}| * |w_{xl+}|} \tag{2.2}$$

ここで w_{xl-}, w_{xl+}は d_{xl-}, d_{xl+}のキーワードベクトルをあらわす。

我々は c_xが小さい部分はその前後で話題の変化が激しく、あるまとまった内容を持つわけではないと解釈し、ある閾値 t で区別することにした。つまり、文書

の c_xを文書の先頭から順に計算すると t 以上の部分と t 未満の部分が交互に出現するが、t 未満の部分に文書を分割するべき場所があると考える。

この方法はキーワードベクトルを比較する上での文書の長さを決められた一定の長さに固定しているので、分割した文書のキーワードベクトルと実際の文書の意味との誤差をを十分小さく抑えることができる。また、文書を一回通して評価するだけでよいので $O(|D|)$ 時間の計算時間となり実用上問題ない。

2.3 キーワードベクトルを用いた文書間の関係の発見

この節ではキーワードベクトルを用いて文書間の意味的な関係を発見する方法について述べる．文書集合 $\{d_1, \cdots, d_n\}$ の 2 文書 d_i, d_jに対して，計算したキーワードベクトルを w_i, w_jとすると，これら 2 つのキーワードベクトルを比較することによって 2 文書 d_i, d_j の関係を近似的に求めることができる．ここではキーワードベクトルの比較方法として前述の文書の分割で用いた方法と同じくベクトルの角度の余弦を求める方法を用いている．つまり 2 文書のキーワードベクトル w_i, w_jに対して計算されるベクトルの余弦

$$c_{ij} = \frac{w_i \cdot w_j}{|w_i| * |w_j|}$$

の大きい文書ほど内容上の関係が深いと考える．また関係の深い文書についてベクトルの成分を比較することによって，どういう意味でその 2 文書が関係が深いのかを分析することができる．この方法を特に文書集合を前節で説明した分割された文書について行なうことによって，文書全体を線形に読むだけではなく意味上の関係をたどることができ，読者は文書の重要かつ必要な部分を早く発見し理解することができるようになる．

2.4 関連研究

本研究で提案する手法と関連の深い研究のひとつに、TextTiling[2] がある。TextTiling では、文章を分割する際に本研究と同じくキーワードベクトルの角度の余弦を計算することによって、文書を複数の節に分割している。本研究の位置付けとして、TextTiling がキーワードベクトルを求める単位をまとまった数の token

であるのに対し、我々は会話文を扱うということに着目しそれを構成するまとまった数の文を単位とした。また、TextTiling では重点を置いてなかったキーワードベクトルを構成するキーワードの選定にも、注意を払った。これによって、文書分割の手法の設計に関する文書への拡張を図った。TextTiling では、本研究で提案する文書間の関係については触れられていない。

第 3 章

キーワードベクトルを用いた文書の構造化実験のための考察

3.1 実験の目的

本研究で提案する手法によって、設計者に理解しやすい形で構造化された設計議事録から、設計の流れを読みとることができるかどうかを検証するために実験を行なう。

そのために、第 2 章で述べた理論を実際の設計議事録であるプロトコルデータに適用する。プロトコルデータとは、複数の設計者が設計中に考えたことを全て声にだしたものを録音する方法（プロトコル解析）によって得られたデータをテキストデータ化したものである。

実際のプロトコルデータの一部を図 3.1に示す。

計算機を用いた実験を進める上で、大きく 2 つ段階、

1. キーワードベクトル法を利用してプロトコルデータをブロックに分割.
2. 分割したブロック間の関係を発見する.

に分けて実験を進めていく。以下で、それぞれの段階について詳細に述べる。

3.2 キーワードの選定

まず最初に、第 2 章で述べたキーワードベクトル法を利用するためにキーワードベクトルを構成する単語をプロトコルデータから抽出・選定する。キーワード

657 J do we have any information that would tell us what's a reasonable weight for the product
658 K we have er
659 J the market testing
660 K what about um bike rack literature do we have anything on existing bike racks?
661 X Yes we have some er comparable products
662 K our goal should be to get all the information out of this little notebook (laughter)
663 I that's what takes longer
664 K mm mm
665 X this is information on Blackburn products
666 I oh OK
667 J (unintelligible) thinking in my mind also but er
668 K this looks a lot like the little backpack frame doesn't it
669 I yeah ..you see we've been it seems like mentally tryin to just because of a similarity in size and shape between the two thinking of ways to er use the same product for the same thing but I dunno that we necessarily I mean we're on a target for fiftyfive dollars I mean if they're able to make that for fortytwo ninetyfive
670 K forty
671 I and if we just add a plastic part
672 J I mean maybe if you look at if you look at this well this is only a buck quarter
673 I yeah yeah OK
674 J yeah these are the panniers but if this was on the rear fork instead of the front fork if you just hang this wire form structure
675 I oh right
676 J and then y'just had clip points plastic clip points that just sort of snap around the frame
677 I mm mm
678 J over the tube of the frame
679 I now my er

図 3.1 プロトコルデータの一部

ベクトルを用いてプロトコルデータ中に含まれている話題内容を捉えるためには、話題中で出現頻度が高いキーワードを選定する必要がある。今回用いたプロトコルデータは、自転車にバックパックを接続するための荷台の設計に関しての会話文である [4]。このプロトコルデータは同一課題に対して異なる被験者が行なった２つのデータがある。以下、その１つを ijk データ、２つ目を fgh データと呼ぶことにする。(名称は被験者の名前に基づく)。まず、１番目のデータ ijk データを用いて以下の作業を行なった。話題内容を捉えられるように、キーワード選定の基準として以下の単語, 熟語に注目し、収集した。

- 自転車、バックパック、接続部品などの物の名前
  - 理由:設計対象物に関する物の名前は、会話をしながら設計作業をする場合、話題の枠組を表すものであるので出現頻度も高く、キーワードとして最も重要である。
  - 例：handlebar,kick stand,wheel,rear wheel,backpack,strap,Velcro
- 設計の際の専門用語
  - 理由：設計対象物間の関係や対象物の性質を表す単語は、設計中の会話において物の名前ほど出現しないが、その単語に関連するところが話題の中心であるので重要である。
  - 例：specific gravity,polypropylene,centimatre,reaction
- 設計作業の際に重要と思われる語や一般的な語
  - 理由：設計者はユーザが設計対象物を使用する場合のことを考えるので、そのための単語は設計中にある程度の頻度で出現する。
  - 例：quick release,camping

名詞を中心に副詞、形容詞、動詞も含めて ijk データから全て手作業で収集した。その結果、合計で約 500 単語になった。これらのキーワード全てを使いキーワードベクトルを構成する。

3.3 ウィンドウサイズの設定と角度データの算出

第 2 章で述べたように、キーワードベクトルが文章の内容を示すものだと捉え、2 つのベクトル間の角度が 2 つの文章の内容の差異の程度を表すという理論の基に ijk データを処理する。この理論ではキーワードベクトルを計算する文章の長さ（ウィンドウサイズ）を変更すると、出現するキーワード数も変化すると予想され、その結果 2 つのキーワードベクトルおよびベクトル間の角度も変化する。話題の分割点を求めていく際に、分割点の前後で隣あった 2 つの一定の長さの文章が、それぞれ内容の異なったまとまりのある話題をあらわす様にするためには、ある適切なウィンドウサイズを定めることが必要である。

そのサイズを定める際に参考になるものとして、人間が ijk データを読んで意味あるものと見なして区切った文章のサイズがある。しかし、会話文を人間が読んで区切ったので当然それぞれの意味ある文章のサイズは一定ではない。そこで、計算機の処理時間、利便性を考え、計算機で意味あるブロックに分割する手段として、ウィンドウサイズを人間が読んで区切った文章の平均サイズあたりを用いるのが妥当ではないかと考えた。常に使える手段ではないが、便宜を図った。人間が分けた文章の長さの平均は約 25 行であった。この妥当性は後の節で述べる。

3.4 算出された角度データ

本実験の処理過程で角度の取り扱いを容易にすること及び角度の可視化のために、角度に対して余弦を計算することにした。

まず、ウィンドウサイズを 25 行に設定して、ijk データ全体に対して求めた角度を図示したものを図 3.2に示す。

x 軸が 2 つのウィンドウの区切り目を示す行番号を表している。y 軸は余弦の値である。全体を見ると余弦の大きさを示す線の連続が山谷の様に現れ、余弦の変化が見てとれる。

本研究の手法では、2 つのウィンドウ間のキーワードベクトルの角度が大きい、つまり余弦の値が小さいところが、2 つの文章間で話題が変化しているところであると見なす。逆に、全体を見て余弦の値が大きいところはその行番号で隣あう両側のウィンドウ内の文章間で話題の変化が小さいところであるといえる。それ

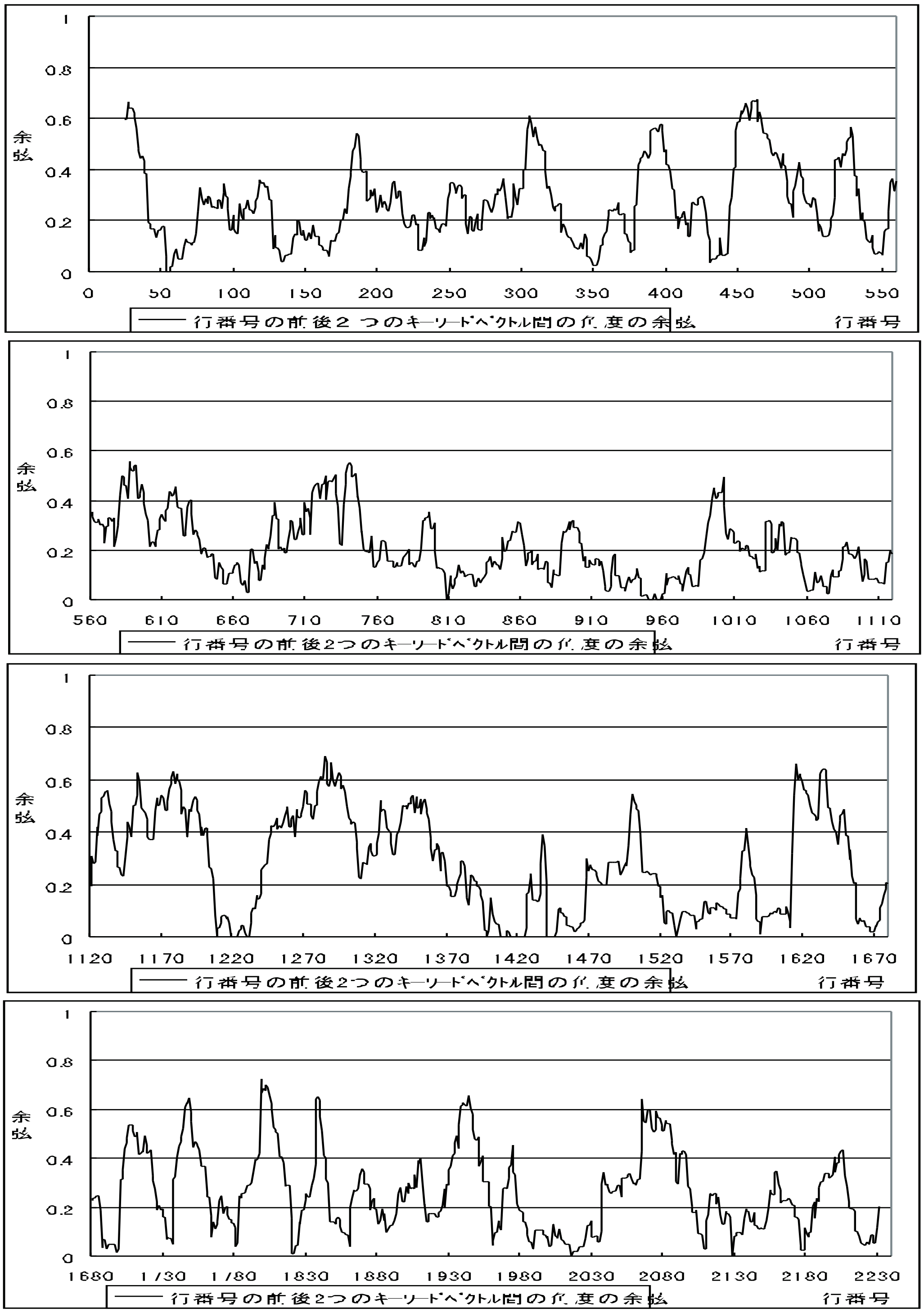

図 3.2 ウィンドウサイズを 25 行にした時の、各行の余弦

が連続していてる部分はあるまとまった内容についての会話部分と捉えることができる。

また細かくグラフを見てみると、大きい余弦の値の連続で山を構成している途中に局所的に谷状に余弦の値が変化している部分も存在する。この部分は大きくまとまりのある話題の中において細かく話題が変化している部分であることを示している。また、山を構成する余弦の値が急激に大きくなっている部分も存在する。その部分は、まとまった話題の中においてさらに特定の事柄について話し合っている部分だと考えられる。

3.5 ウィンドウサイズの違いによる余弦の比較

前節で人間が ijk データを 87 の文章に区切った時の区切られた文書の平均行数 25 をウィンドウサイズに設定した。その妥当性を検証するために、ウィンドウサイズを様々な値に変更しそれぞれの場合の角度データを求めて結果を比較する。

具体的にウィンドウサイズを 5,15,20,35,50 行にして、それぞれの場合の角度データを比較してみた図 3.3。

ウィンドウサイズの違いによる、余弦データの特徴を見てみる。

- size5

  余弦の値が、0.0000000 あるいは 1.0000000 のところがかなり出現している。これは、サイズが小さいので、同じ内容の話題の部分を 2 つに分けたり (1.0000000)、全体としてはまとまった内容の中のある特定の部分だけ出現するキーワードが激しくに変化する (0.0000000) が出てくるのが理由と考えられる。余弦データ全体を見て意味あるブロックと捉えられる部分が少なく、機械的に取り出すのは難しく思える。

- size15

  size5 に比べて 0.0000000 あるいは 1.0000000 がかなり減少した。size5 の場合の問題点が減少したからだと考えられる。ブロックとして、人間が捉えやすい部分が出現してきたが、全体的に各行ごとの余弦の値の変化が激し

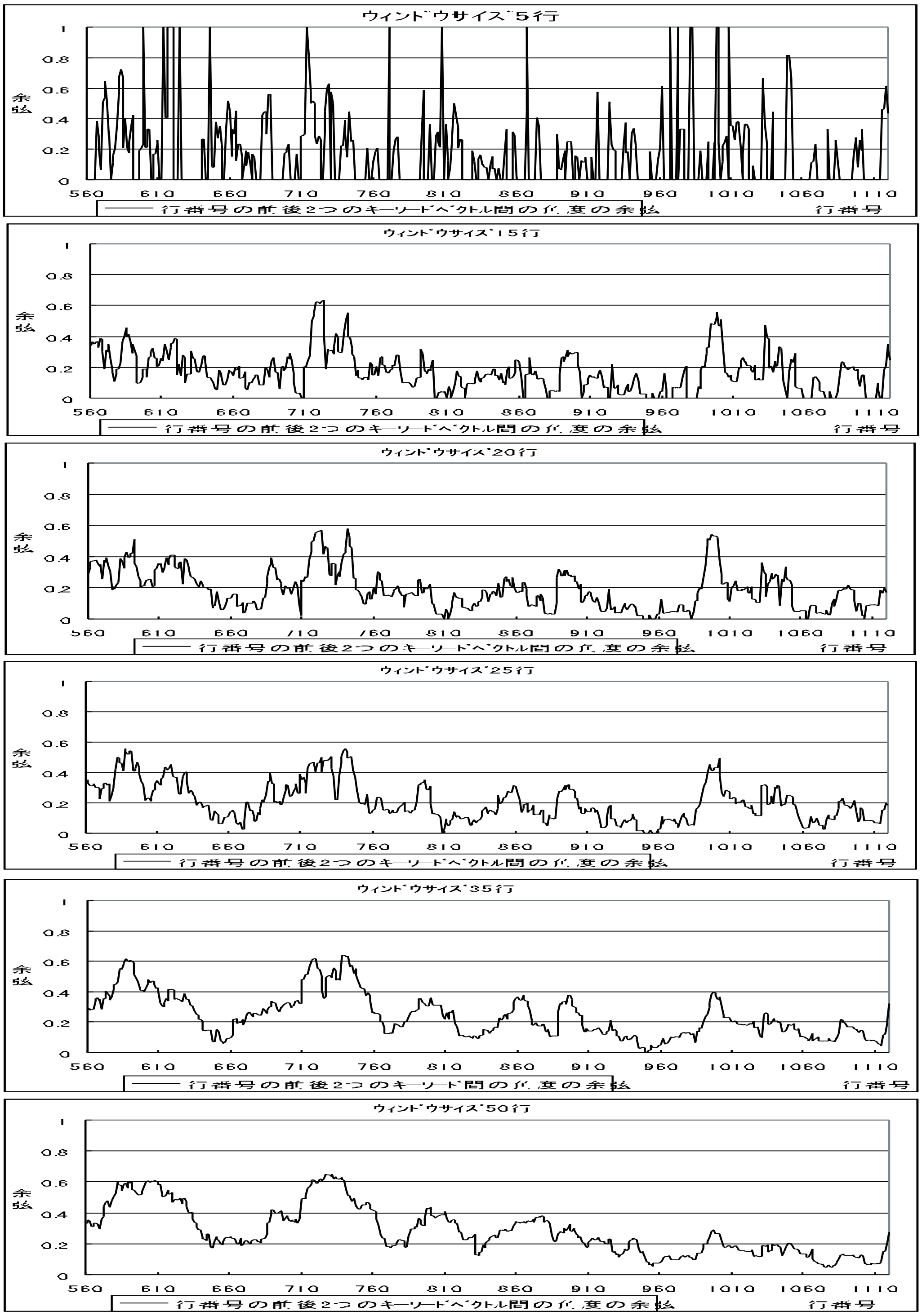

図 3.3 ウィンドウサイズの違いによる結果を比較

く、まとまった内容を示すブロックとして機械的に取り出しにくいと思われる。また、サイズが小さいと思われる。

- size20

  size5 に比べてかなり 0.0000000 が減っている。急激な余弦の値の変化が減少してきたおかげで、ブロック全体が滑らかな形を描くようになってきて、ブロックを捉えやすくなっている。size25 と形が類似している。

- size35

  size25 に類似しているが全体的に余弦の値の変化が少なくなってきていて、size25 に比べて、ブロックとして捉えにくくなっている部分も現れている。

- size50

  size35 と比較してかなり余弦の値の変化が減少してきている。全体的にフラットな形になっているので、ブロックとして捉えやすい部分が、減少している。これは、ウィンドウが大きくなるほどその中に複数の話題を含んでいる確率が高くなりある程度詳細な話題の変化を的確に捉えることが難しくなるということを示している。

各サイズを比べて見た結果、人間が意味あるブロックだと認識しやすく且つ機械的に処理しやすい滑らかな形を持つ角度データを表示しているサイズはおよそ size20∼size35 の範囲内のグラフである。その間でどのサイズを採用してもサイズの変化に対する余弦データの変化が小さい。したがってウィンドウサイズとして 25 を採用することは妥当だと思われる。従って、この平均の値 25 行をウィンドウサイズに設定し、ijk データの先頭から 2 つのウィンドウを設ける。つまり行番号 1 ∼ 25 と 26 ∼ 50 にウィンドウを配置しその範囲でそれぞれキーワードベクトルを計算して、2 つのキーワードベクトルの角度を求める。この角度を、25 行目で区切った時の前後の文章の話題内容の差異とする。次にウィンドウをそれぞれ 1 行づつずらして、2 ∼ 26 と 27 ∼ 51 で同じようにキーワードベクトル間の角度を求める。この処理を ijk データの末尾まで行なう。また、ここでは最初の 24 行には、話題に変化がないものとして捉えることにする。

3.6 閾値の設定

設計者の設計対象に関する、

- 話題の中心を複数含む大きくまとまった意味を持つ話題
- 話題の中心
- プロトコルデータ全体を 50 の内容に分割した時の 1 つの話題

などの情報を得たいという要求に応えることが出来るように、前節で求めた余弦データをそれぞれの要求に応じた意味あるブロックに分割することを考える。

プロトコルデータから、大きくまとまった意味を持つ話題を得たいと思えば、全体を見て余弦の値が小さいところを話題の変化する部分として捉え、そこで分割した結果得られるブロックを大きくまとまった意味を持つ話題のブロックとすることができる。また、話題の中心をとり出す場合には、余弦グラフ中に局所的に出現する谷の部分を詳細な話題の区切り部分と評価して、プロトコルデータを分割する必要がある。分割数を指定した場合は、余弦グラフ中からブロックとして捉えやすい部分を分割数分選ぶ必要がある。以上の条件を満たすような余弦データを意味あるブロックに分割する手法として閾値を使用することにした。

閾値以下の値をブロック切りだしの対象外とするとする。

閾値を低く設定すると除外余弦の値は小さいものになり、より話題が大きく変化しているところで分割することになるので、小さな話題を複数含む大きくまとまった意味を持つ話題ごとにブロックが分割できると予想できる。

逆に、閾値を高めに設定するにつれて、余弦データ中にある極小値を小さいものから順に分割部分と見なせるようになり、詳細な話題についても一つのブロックとして捉えることができる。その場合、一つのブロックの大きさも小さくなってくる。

上記のことを踏まえ、閾値とブロックの数との関係を調査することにした。計算時間、閾値の変化に対するブロックの変化度合などを考慮に入れると、閾値を 0.000 から 1.000 まで 0.001 単位で設定しそれぞれでブロック数を計算することが妥当ではないかと考えた。

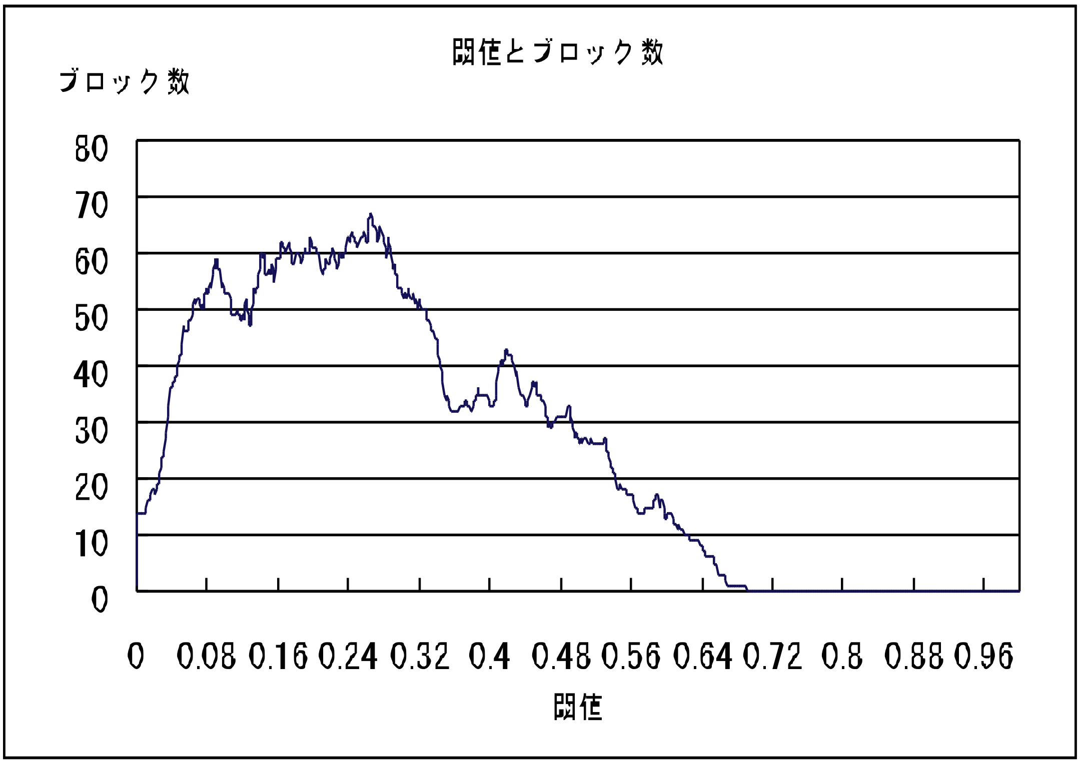

図 3.4　閾値とブロック数の関係図

また、ブロック数としてカウントするのは、会話を構成する最小の単位（二人）を考慮に入れて、余弦データが 2 行（一人 1 行）以上ある部分とした。

図 3.4は、閾値とブロックの関係を示したものである。縦軸が分割されるブロック数、横軸が閾値を表す。

- 閾値が小さい時、およそ閾値の変化に対して分割数は増加する。

  閾値が十分小さい時には余弦の値が閾値よりも大きい部分が多くなり、閾値によって分割点と見なされる部分が少なくなり、結果的に分割されるブロックも少ない。

  閾値が大きくなるにつれて、閾値近辺の余弦グラフ上の極小値地点が文章の分割点として新たに評価される数が、閾値によってブロックとして見なされなくなった小さいブロックの数よりも多くなるので、総ブロック数は

増加する。

- 分割数がピーク近くになると閾値の変化に比べて分割数は安定している。

  ある範囲内で閾値を変化させても分割数がピーク近くでは、分割数が変化しない。これは、その範囲内でのブロックの形状が閾値の変化に対して、比較的安定したものであるということを示している。閾値が変化してもブロックの両端が削られて大きさが小さくなる程度の変化でしかない。主要なブロックと思われるものはすべて捉えられたといえる。

- 閾値をある一定以上にすると分割数は減少する。

  閾値の増加に伴って、ブロック内最大の余弦の値がそれほど大きくないブロックが閾値以下になってきて、余弦の値がかなり大きいブロックしか残らなくなってくる。

図 3.4から、ほぼ主要なブロックに分割できたと思われる閾値 0.163 を余弦データを意味あるブロックに分割するための閾値に設定する。

3.7 オントロジーの構築

第 2 章で述べたように、あらかじめ抜きだしたキーワードを基にキーワードベクトルを計算し文章をブロックに分割する。しかしこの手法のみだと、文章中で本来なら分割を望まない部分を分割部分と見なす危険性も存在する。

例えば、「ある部品どうしをつなげる部品について」というまとまりのある話題があった場合、その文章中に内容に関連するキーワードとして bolt のみが出現したのなら問題は無い。しかし、キーワードとして nut が選ばれていて、それが文章中に bolt と共に出現していた場合、キーワードベクトルによって一つの話題として捉えられずに、途中で分割される危険性が出てくる。

このような危険を回避するために、“概念的に良く似たものをまとめる” という考え方を採り入れる必要がある。例えば、上記の例の場合 bolt,nut は、screw(ネジ)という概念でまとめておく。つまり、bolt と nut が同時に出現しても、screw

が出現したと解釈が出来るようにしておく。そうすれば、screw についての話題であると計算機は捉えることができ、望まない分割を避けることができる。

さらに、キーワード botton も出現する「二つのものを留めるための器具について」のような話題を間違って分割したくない場合には、botton,screw を "ものを留めるための器具" という概念でまとめておけばいい。

このように世の中に存在するものを、ある概念がそれをより抽象的に表した概念の意味を含んでいる形で捉えたもの、つまり概念の背景にある構造を記述したものをオントロジーと呼ぶ。

このようなオントロジーを利用することによって、意味のあるブロックを間違いなく切り出す可能性を向上させることが出来ると思われる。そこで、オントロジーを今回用いた ijk データのように設計に関する話題に適用するために、『設計に関するオントロジー』構築を試みた。

今回構築するオントロジーの機能を考え、フレームに基づくオントロジー [7] を準備した。

概念はそれが文章上で表れる表現（複数可）を持つ。また、概念間は抽象-具体関係を持つ（複数可）。

オントロジー構築に当たって、抽象的な部分と、具体的な概念の部分とで別々の方法を用いた。

- 抽象的な部分

  概念が抽象的になるほど設計者自身の思考のみでオントロジーを構築することは難しくなる。そこで、この部分を構築するためには、抽象的な概念構造を詳しく収録したＥＤＲ概念辞書を利用することが効率的であると考えた。ＥＤＲ概念辞書を見ながら手作業で概念構造を構築した。

- 具体的な部分

  概念が具体的になるほど設計に関してより専門的な単語、そのオントロジーの分野を特徴付ける単語が増加してくる。この部分は、本実験で選んだキーワードが概念として位置するところである。専門分野の概念関係に詳しい設計者自身が構築する方が具体的な概念に関して詳しく収録していないＥ

ＤＲ概念辞書を利用するよりも効率的である。設計者自身の思考のみで手作業で構築した。

設計に関する概念体系は、キーワードを含んで約 1100 概念から構成されたものとなった。今回構築したようなオントロジーと類似したものにシソーラスがある。今回構築した設計に関するオントロジーは、設計に特化したシソーラスと形は同じになっている。しかし、オントロジーは、それを構築する人間の視点が深く入る。設計者によって構築するオントロジーも変化してくるであろうし、同じ人が同じものを設計するにしても視点が異なればオントロジーは変化してくる。このようなものに対して、シソーラスのような一般性が求められるようなものでは対応し切れない。図 3.5は、我々が開発した設計者支援環境である Designers Amplifier システム [3] におけるブラウジング機能を利用して表示させたものである。アイコン一つ一つが概念を表していて、アイコン→アイコンが上位概念→下位概念を表している。

3.8 オントロジーの利用方法

今回、ある概念は一つの意味しか持たないという方針で構築に当たった。ijk データ内から選出したキーワードは、ijk データ内では１つの意味しか表していないというのがその理由である。しかし例えば、rack という単語は今回のプロトコルデータ中では自転車の荷台の意味でしか使用されていないが、本来なら戸棚という意味もある。従って、多義語の存在も考慮に入れてのオントロジーの利用方法について述べる。

キーワードベクトルは、本文中に出現するキーワードによって構成されるが、そのなかのあるキーワードをオントロジーを利用してある概念でまとめた場合、キーワードの代わりにその概念を概念ベクトルを構成する要素として扱う。

先述した例を用いると、nut,bolt は、screw でまとめられるので、キーワードベクトルを構成していた nut,bolt の部分は、両方とも screw に置き換わり、その結果概念ベクトルの構成要素が一つ減少することになる。この場合、本文中に nut と bolt がそれぞれ 1 回ずつ出現したとすると、screw が２回出現したと計算する。

オントロジーの構造を反映した詳細な出現回数の計算方法を以下に示す。

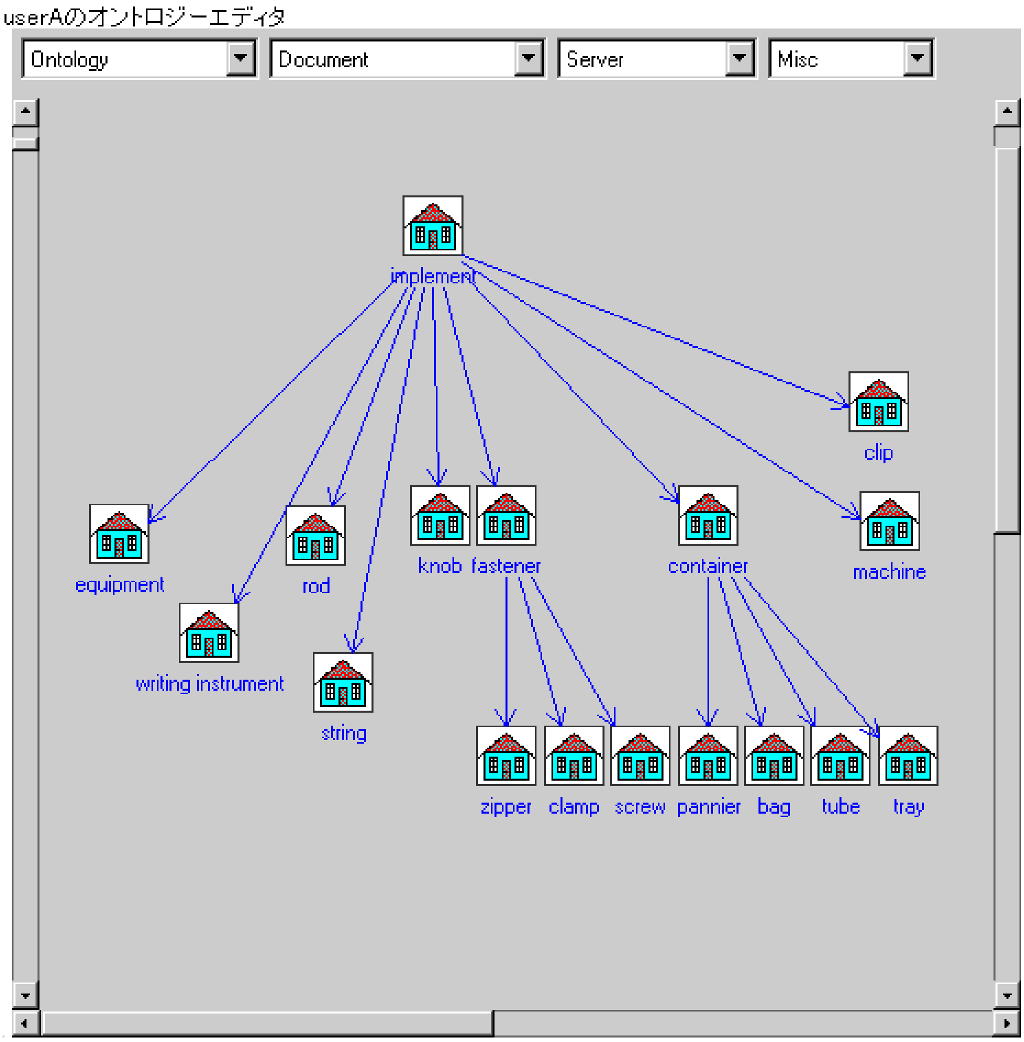

図 3.5 自転車の荷台の設計に関するオントロジー

具体的な概念は抽象的な概念に対して、意味を継承している割合つまり重みを計算する。

- 単一継承

  重みを 1 と設定する。

- 多重継承

  具体的な概念が、意味を継承している抽象的な概念数だけ重みを分割する。2つの概念の意味を継承している場合には、それぞれの概念の重みを $0.5(1/2)$ と定める

具体的な例を示すと、

- 単一継承の例

  親概念 x に子概念 a, 子概念 b が含まれていて、ウィンドウ中の本文に子概念 a が 1 回, 子概念 b が 2 回出現した場合、親概念 x は、$1*1$(キーワードから見たまとめる概念の重み)$+2*1=3$ 回出現と計算する。

- 多重継承の例

  親概念 x と親概念 yに子概念 a,b が含まれていて、ウィンドウ中の本文に子概念 a が 2 回出現した場合、親概念 x は、$2*0.5$(キーワードから見たまとめる概念の重み)$=1$ 回, 親概念 yは、$2*0.5$(キーワードから見たまとめる概念の重み)$=1$ 回出現したと計算する。

- 多義語

  多義語が持つそれぞれの意味の上位概念に対して、単一継承、多重継承を計算する。例えば、ある概念 x が、p と qの意味を持っていた場合 p の意味としての親概念が P、qの意味としての親概念が $Q1$,$Q2$ であったとする。本文中に x が 2 回出現したとすると、Pが 2 回、$Q1$ が $2*0.5$=1 回、$Q2$ も $2*0.5$=1 回出現したと計算する。

計算機で上記の計算を行なうことができるように。

- （キーワード、まとめる概念、キーワードから見たまとめる概念の重み）

の組をオントロジーから求めるようにした。

オントロジーを利用して ijk データから意味あるブロックを抜き出す実験として、キーワードをオントロジーにおいて全て 3 段上位に位置する概念でまとめることにした。その理由として、

- 3 段上位でまとめると概念ベクトルを構成する要素が約 1/5 となり効果を検証しやすい。
- 個別に意味を考えながら取り扱うには時間と労力が必要。

ということがあげられる。

キーワードと 3 段上位の概念との関係を表 3.8 に示す。

この関係図を基に概念ベクトルを構成し、プロトコルデータから余弦データを求める。

3.9 出現頻度を基に半自動的にキーワードを選出

設計者が行なうキーワードの選定の労力削減の可能性を探るために、プロトコルデータを構成している単語から出現頻度に着目してキーワードを選び出す方法も試みる。統計的なキーワード抽出の手法として $tf \cdot idf$法があるが、この方法は、分割された複数のブロックが存在する場合に重要語の選定に役立つが、今回のブロックを切り出す際には適切な手法ではないと判断し使用しなかった。

具体的には、純粋にプロトコルを構成している単語を全てとりだし、そこから

- 冠詞 a,the
- 代名詞 they,it
- 前置詞 at,on
- 動詞の中でかなり一般的な語、助動詞 have,is,can

表 3.1　キーワードと概念と重みの関係

キーワード	3 段上の概念	重み
rear	前後左右	1
rail	implement	1
rack	形で捉えた物	0.5
rack	器具	0.5
pole	形で捉えた物	1
plug	開閉にかかわる器具	1
plan	idea	1
pack	具体物を持ち運ぶための器具	1
part	部分	1
top	部分位置	1
nut	implement	0.5
nut	接続具	0.5
net	形で捉えた物	1
map	図表や絵を主とする情報	1
lug	a part of a physical object	1
guy	パーゾナリティで捉えた人間	1
bud	対人関係で捉えた人間	1
bed	physical object	1
bar	形で捉えた物	1
bag	implement	1
zinc	物質の成分	1
wide	形の値	1
advantage	性質	0.33
advantage	物事に対する評価	0.33
advantage	他と内容的比重の異なる部分	0.33
walk	体の動作を行う	0.5
walk	方向の決まっている移動	0.25
walk	主体の移動	0.25

などを取り除いたもの全てをキーワード候補とした。キーワード候補数は ijk で 1600 であった。

次にそこから手作業で選出した数とほぼ同数の上位 500 個取り出し、そのキーワードからキーワードベクトルを構成し、ブロックの切り出しを試みた。その結果と、手作業で選出したキーワードから得られたブロックとを比較する。

第 4 章

単数プロトコルデータを用いた実験

4.1 人間がプロトコルデータを分割

本研究では、計算機によるプロトコルデータの分析・構造化が目標である。計算機による本研究の手法の効果を確かめるために、人間がプロトコルデータを読んだ後、その中でまとまった話題と思われる部分ごとにプロトコルデータを区切る。その結果区切られた文章は人間が考えて区切ったものであるので、それぞれが意味のある文章だと言える。計算機を利用して分割するブロックも、“意味のあるブロック” として切りだす。双方を比較することが、計算機によってプロトコルデータを構造化した結果を評価する手段の一つとなり得る。以下では、人間が区切った文章にそれぞれタイトルを付けたものに「 」で示した。

4.2 キーワードベクトルと閾値によって切り出されたブロック

閾値を 0.163 に設定したとき分割されたブロックは図 4.1のようになっている。x 軸が行番号で、y 軸が余弦、グラフ中の縦線が人間が区切った部分を示している。図 4.1からは以下のようなことが分かる。

- 切り出されたブロック

  分割されたブロックをそれぞれを見てみると、2,3 行からなる部分も含まれているが、ある程度の大きさを持つブロックを切り出すことが出来ている。また、プロトコルデータ全体においてブロックが集中しているところも少なく、適度に分散してブロックが切り出されている。

- ブロック以外の部分

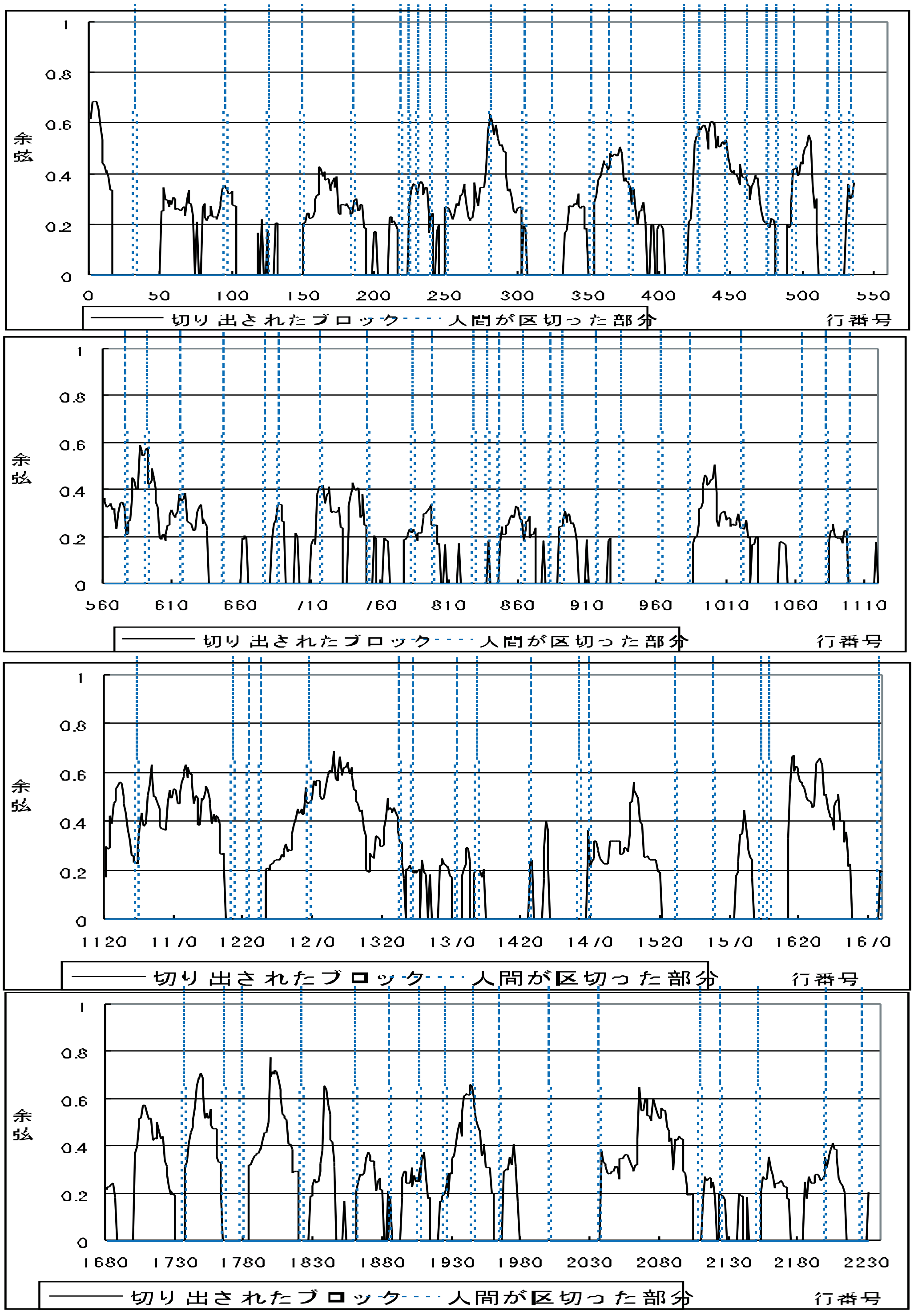

図 4.1 切り出されたブロックと人間が区切った部分

ブロックとして切り出された部分以外のところは、分割線がその中にあるということを示しているが、その部分がかなり存在する。プロトコルデータ全行数の約半分を占める。

- 人間が区切った部分との比較

  計算機を利用して切り出したブロックと人間が区切った区切り線の位置とを比較してみる。プロトコルデータの前半部分は、複数の区切り線に渡る大きさのブロックが切り出されている。また、区切り線を超えて切り出されている部分もある。後半部分は切り出したブロックがちょうど区切り線の中に収まっている部分が多い。人間が意味あるまとまりと解釈して区切った文章でも、機械による意味あるブロックに切り出されていないところも多くはないが存在する。

以上のような特徴を持つブロックについて、本研究が提案する手法の有効性を検証するために以下ではいくつかのブロックを例にとって、人間が分けた文章と比較し、そのブロックが切り出された理由を分析する。

- ブロック 360 ∼ 373

  人間は 349 行目から 373 行目までを一つの文章としている。人間がつけた文のタイトルは「取り付ける材料の重要性」である。このブロックの文章中で我々がキーワードベクトルを構成する (キーワード 出現回数) は、(joining 4),(material 6) である。

  このブロックは人間と同じように文章を捉えた例である。

- ブロック 444 ∼ 505

  人間は 442 行目から 505 行目までを 5 つの文章に区切っている。人間がつけた文のタイトルはそれぞれ、自転車とバックパックを固定する方法について

  「マジックテープ」「金具」「革ひも」「カバンを介す」「チャイルドシート」

  についての話題である。キーワードベクトルを構成するキーワードの出現回数は、それぞれ

rack	5	3	2	4	0
clip	0	1	0	0	0
bag	0	0	0	4	0
strap	5	0	2	3	0
child seat	0	0	0	0	1
backpack	0	0	0	3	2
pack	4	0	4	0	0

である。

この部分は、人間が文章内容の細かい変化を読みとり文章を区切った。それに対し、計算機の処理では、うまく分割できなかった。それは、ある文章の意味を表すキーワードがその文章に出現する他のキーワードの出現回数よりも少ない場合、キーワードベクトルがそのキーワードを文章の内容を表すものだと捉えにくくなり分割点として判断できない場合が出てくるというのが理由と考えられる。あるいは、文章の区切り間隔が短い文が連続した時、ウィンドウサイズがその間隔よりも大きくなった場合にキーワードベクトルが短い話題の変化を捉えにくい場合が出てくるというのも理由になる。しかし、切り出したブロックの表す意味は「バックパックと自転車との接続方法の議論」であり、5 つの文章を合わせた意味と相違はない。

- 文章 2001 ～ 2036

  この人間が区切った文章の内容は、「盗難防止のロック機構について」であるが、全体的に余弦の値が 0 に近く、ブロックとして切り出すことが出来なかった。これは、一つの話題のなかに出現するキーワードの種類がかなり多く散在しているので、キーワードベクトルを利用すると話題の区切り目の連続だと捉えてしまった結果であると思われる。出現するキーワードは、pivot,wing nut,pack,snap,clip,strap,srew,frame,lock,weight などである。

プロトコルデータ全体で、切り出されたブロックの容を見ていくと、人間が区切った文章と等しいとはいえないが、それぞれのブロックはまとまった意味を持つ場合が多かった。

4.3 オントロジーを用いた結果

オントロジーを用いた結果得られた余弦データは次の特徴を持つ。

- 全体的にオントロジーを使わないデータよりも余弦の値が大きい。

オントロジーを利用してキーワードをまとめた結果、概念ベクトルがここのキーワードの違いに対してあまり敏感でなくなる。

その結果、概念ベクトル間の角度を示す余弦の値も大きくなったと考えられる。

次にこの余弦データを意味あるブロックに分割し、オントロジーを利用する場合と利用しない場合との結果の違いを発見する。その際、オントロジーを利用した余弦データを、閾値と分割ブロック数の関係のグラフから適当に求めた閾値を利用してブロックに分割してそれを比較するという方法も考えられる。しかし、両者は余弦の分布が違うので直接比較するよりも、正規化した値を出してそこから違いを読みとった方が、より正当に評価できると考えた。

図 4.2は、オントロジーを利用した場合と利用しない場合とで得られた各行の余弦を余弦全体の平均と標準偏差から正規化したときの値を示したグラフである。図 4.3は、各行ごとに差を求めたものである。図の系列が 2 種類ある方が正規化したグラフで、1 種類の方が差を求めたものである。

差を示したグラフ上で正の値をとると、オントロジーを利用しない方がオントロジーを利用する場合よりも余弦データとして、よりブロックとして認識されやすい形をとっていると理解できる。逆に、負の値をとると、オントロジーを利用した方が、ブロックとして認識されやすい。

グラフ全体を眺めて見て分かることは、負の位置にあるグラフは正の位置にあるグラフに比べて全体的に鋭角的で、さらにその極値の絶対値は正の位置にある極値よりも大きい。このことから、オントロジーを利用することによって、余弦の増加範囲は狭いが、余弦の値の増加が急激であるので、ブロックとして新たに切り出される部分が出現する可能性が高いと考えることができる。逆にオントロジーを利用することによって、ブロックを構成する余弦の値の減少範囲は広くなるが、余弦の値の減少度合は小さいと考えられる。したがって、従来のブロック

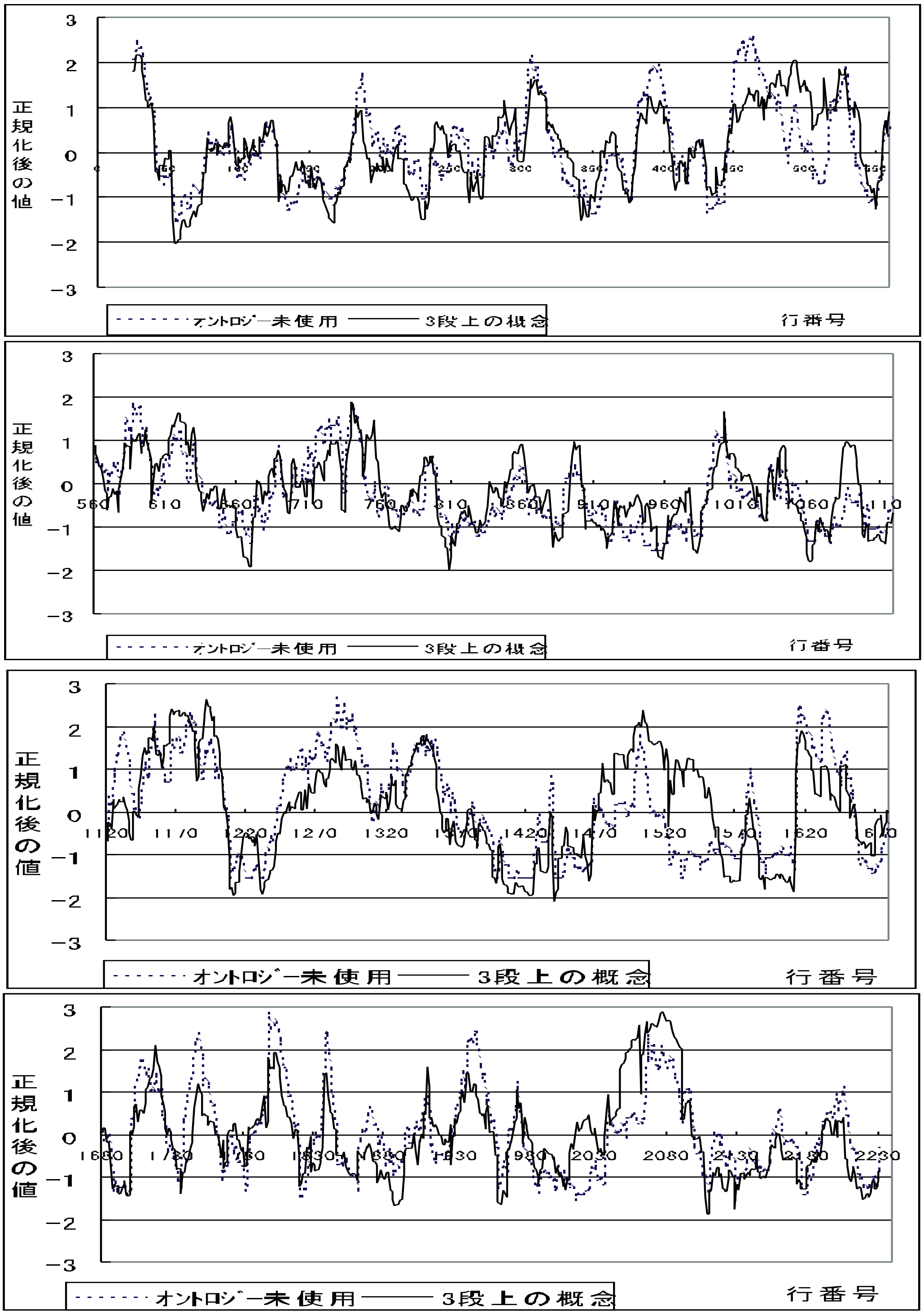

図 4.2　3 段上の概念でまとめた時と、オントロジーを使わない時との比較

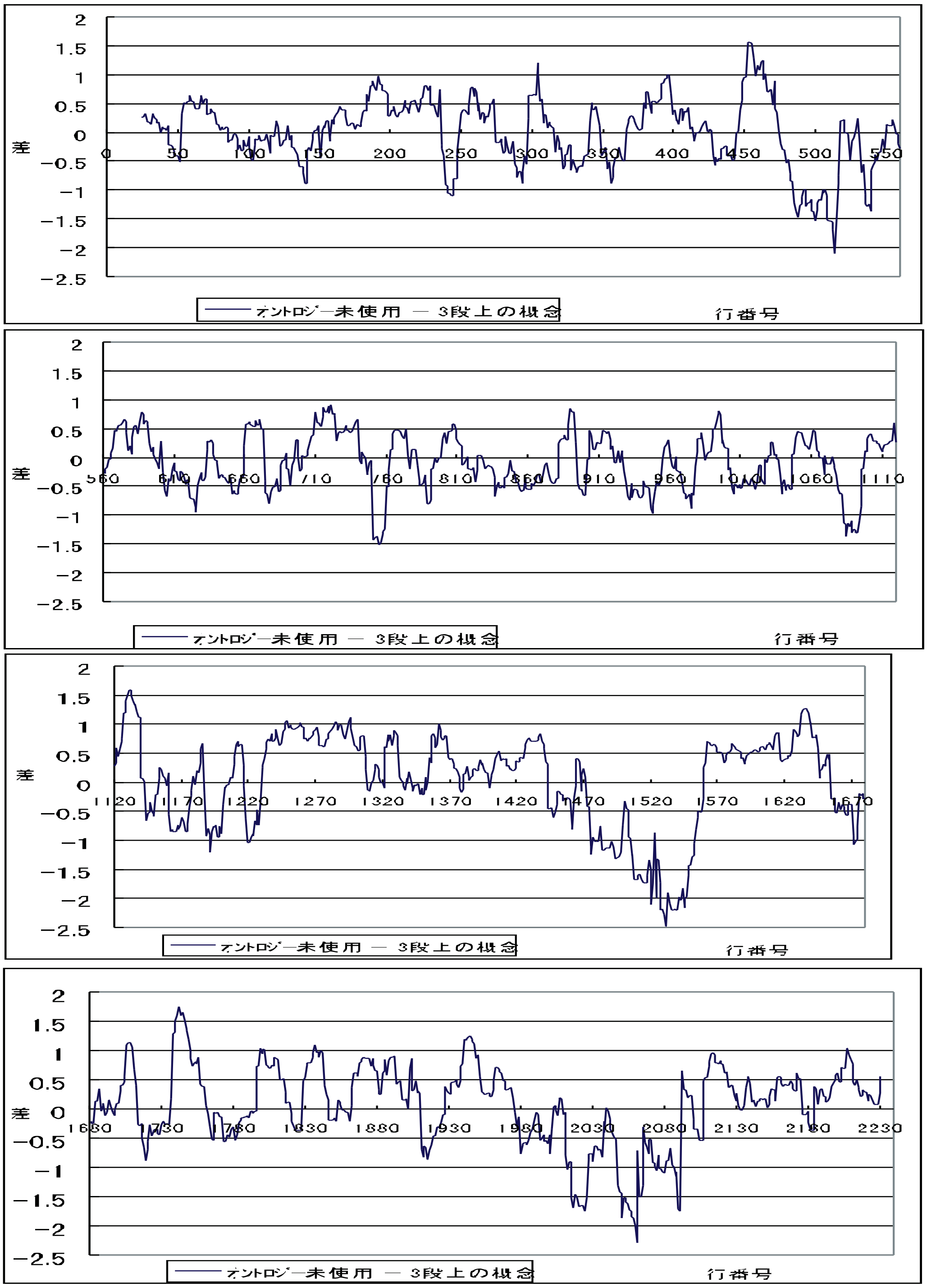

図 4.3　3 段上の概念でまとめた時と、オントロジーを使わない時との差

を形作る山が広い範囲でなだらかなるが、山と認識される部分の減少度合は、小さいと考えることができる。

このグラフから、オントロジー利用した時に余弦値の変化が大きいと思われる特徴を示しているところについて、いくつか例を抜きだして分析する。

- 差分グラフ上で負の位置にある部分。
  - 行番号 1470 ～ 1562

    オントロジーを利用することによって、文章中に出現する (キーワード 出現回数) が (tray 6),(screw 2),(tube 8) が (implement 15) にまとめられた。この行番号の部分は、文章タイトル「トレーの構造について」の「トレーを固定する方法について」の二つの部分とほぼ同じ位置にあたる。詳しく見ると、tray と tube は、両タイトルの文章中にわたって出現するが、screw は後者のタイトルの文章中のみに出現していることが分かる。新たなブロックの切り出しが特に後者タイトルの文章中において起こる可能性が高いと思われる。
  - 行番号 2010 ～ 2038

    オントロジーを利用することによって、文章中の (キーワード 出現回数) がそれぞれ、(wing nut 1)(clip 1)(cable 1)(screw 1) が (接続具 1)(implement 3) に、まとめられた。この行番号の部分は文章タイトル「盗難防止のロック機構について」の位置にあたる。様々なキーワードが一つの話題に対して共通のキーワードとしてまとめられ、新たなブロックが切り出されると思われる。
  - 行番号 476 ～ 518

    オントロジーを利用することによって、文章中に出現する (キーワード 出現回数) が (backpack 6)(pack 4) が (具体物を持ち運ぶ器具 10)、(bag 4)(zipper 4) が (implement 8) にまとめられた結果、一つのブロックとして捉えるようになった。人間が複数に区切った文章を一つにまとめる結果になった。オントロジーによって、細かい変化が捉えられなくなるかわりに大まかな話題として捉えた例である。

- 差分グラフ上で正の位置にある部分。
  - 行番号 1560 ～ 1645

    文章タイトル「バックパックとトレーを結び付ける方法について」、「スケジューリング」、「トレーの詳細について」の部分にあたる。オントロジーを利用しても、文章中に出現する (キーワード 出現数) が (tray 5) が (implement 5)、(strap 7) が (ひも類 7)、(inch 7) が (数量 7) になるなど、文章の意味を表すキーワードについてまとめられなかった。従って、全体的な余弦値の上昇の影響でこれらのキーワードによって出現していたブロックの山が相対的に低くなってしまった。
  - 行番号 1784 ～ 1821

    オントロジーを利用することによって、文章中に出現する (キーワード 出現数) が (inch 6) が (数量 6) に (triangle 2) が (形の種類 2) になるなど、複数のキーワードがある概念にまとまることはなかった。全体的な余弦の値の上昇の影響で、これらのブロックの山が相対的に低くなってしまった。文章のタイトル「rear stay」の部分にあたる。

4.4 半自動的にキーワードを選択した結果

図 4.4は、半自動的に出現頻度順にキーワードを選んだ時、手作業でキーワードを選んだ時の余弦を正規化して比べた図で、図 4.5は各行ごと差をとったものである。

全体的に、手作業でキーワードを選んだときのグラフよりも山谷が明確でない部分が多い。特に後半部は、手作業の場合に切り出された複数の山が大きな一つの山として切り出されたり、またはほとんど山として捉えられない部分が見られる。

新たなブロックの切り出しの可能性がある部分と逆にブロックが捉えられなくなったところについて見ていく。

- 差分グラフ上で負の位置にある部分

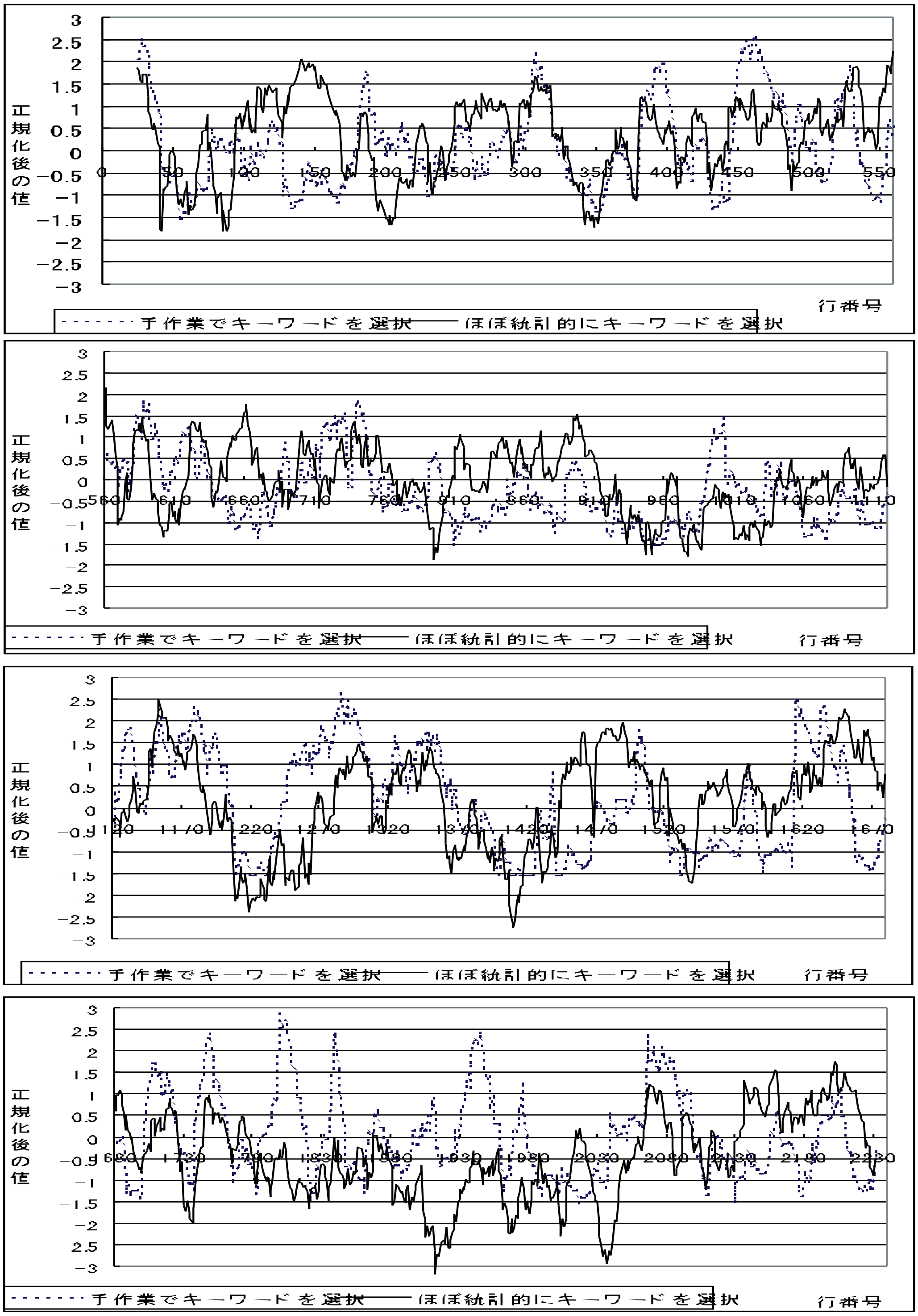

図 4.4 出現頻度順にキーワード選択と手作業でキーワード選択との比較

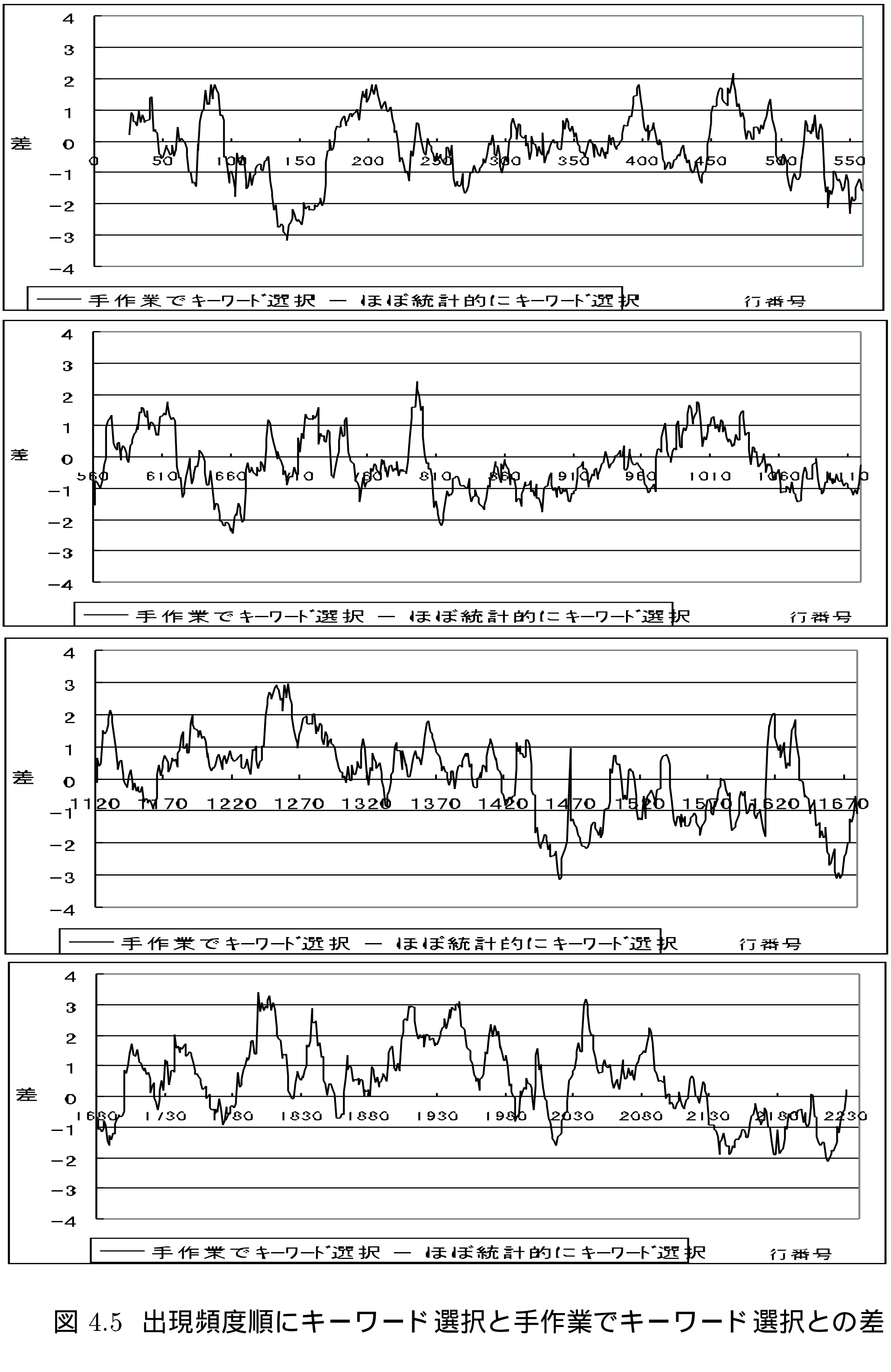

図 4.5 出現頻度順にキーワード選択と手作業でキーワード選択との差

  - 行番号 127 ～ 169

    出現頻度順に処理した結果、この部分に 6 個出現している then という単語がキーワードとして捉えられるようになり、ブロックとして切り出されるようになった。文章のタイトル「バックパックの取り外しの時間を考慮」と「バックパックを wing-nut で留めることについて」のほとんどに位置している。

  - 行番号 1640 ～ 1700

    出現頻度順に処理した結果、この部分に 7 つ出現する two という単語がキーワードとして捉えられるようになり、ブロックとして切り出されるようになった。タイトル「プラスチックの軽さと荷台の大きさについて」の話題であるが、内容と直接関係のないキーワードによって一つの話題と捉えた例といえる。

- 差分グラフ上で正の位置にある部分

  - 行番号 1787 ～ 1820

    「rear stay の高さについて」の部分であるが、細かく見ると短い範囲に (think 2),(forty 2),(plus 2) などの出現頻度順に得たキーワードが出現することとで、本来のキーワードの inch の効果が減少し、細かく話題が変化していると計算機は捉えたと思われる。ブロックとして切り出せない。

  - 行番号 1751 ～ 776

    「bungee strap のコストについて」の部分であるが、ここでもまた OK,twentyfive などが、本来のキーワードである cost,strap の効果を減少させ、細かく話題が変化していると捉えられた例といえる。

出現頻度順に半自動的に選び出したキーワードから新たに切り出されるブロックについて見てきたが、ブロックとそれを切り出す要因となったキーワードとはあまり関係がなかった。また、ブロックとして捉えられなくなった原因がブロックの意味と直接関係のないキーワードによって起こっている。

第 5 章

複数プロトコルデータを用いた実験

5.1 オントロジーの交換

キーワードを基にした本研究の手法を同じテーマの設計者に対して一般的であるか、検証してみる。つまりキーワード、オントロジーの再利用性である。また、オントロジーの違いによって切り出されるブロックがいかに変化するかということも検証する。

我々が実験を進めるうえで使用してきた ijk データとは別に『自転車にバックパックを取り付けるための部品の設計』という同じテーマに関して、f,g,h の三人組で設計に取り組んだ時の 2 つ目のデータ、fgh データが存在する。そのプロトコルデータについて、ijk データを分析した人とは別の人が、文章中からキーワードを約 400 個選定したあと、約 800 の概念を持つ fgh オントロジーを作成した。

その fgh オントロジーを ijk データの分割に際に利用することにした。つまり fgh オントロジーが、過去に構築したオントロジーにあて、ijk データをこれから参照するの新しい設計情報にあてる。

fgh オントロジーの特徴は、中段の概念が少なめで荒く階層構造が作られている。もちろん、fgh で選出したキーワードには、ijk データで選出したキーワードがかなり含まれているが、含まれていないものも数多く存在する。

fgh オントロジーの構造を考慮に入れて、fgh オントロジーを利用して fgh キーワードを一段上の概念でまとめたものと、ijk オントロジーを利用して ijk キーワードを 3 段上の概念でまとめたものとで、ブロックの分割結果を比較してみた(図 5.1,5.2)。

5.2 交換したオントロジーを利用した時の比較

グラフ全体を見ると、前半部では、両者の形状に何ヶ所かで相違が見られるが良く似た形状であると読みとれる。特に会話が始まって 1/4 経った辺りでは、良く似た形をしている。後半部では、両者の形状に著しい違いを読みとることが出来るところも多くなってくる。ijk において、大きな山として捉えられるところは、形状に違いがあるが、fgh オントロジーの方も山を形作っている場合が少なくない。また、ijk オントロジー利用と ijk キーワードのみとのグラフの形状の変化に比べて、大きく変化していると捉えられる。

新たにブロックとして捉えられるところと、ブロックが捉えられなくなった部分をとりだして分析してみる。

- 差分グラフ上で負の位置にある部分
  - 行番号 751 ~ 776

    fgh オントロジーを利用することによって文章中の (キーワード 出現数) は (tube 1)、(telescoping tube 1)、(downtube 2)、(seat tube2) が概念 (tube 6) にまとめられた。この部分は、文章は「ラックの形を自由に変えられる仕組みについて」とほぼ同じ位置にあたる。ブロックとして切り出せるようになった部分である。
  - 行番号 260 ~ 281

    fgh オントロジーを利用することによって文章中の (キーワード 出現数) は、(lock 4) が概念 (開閉にに関わる器具 4) でまとめるようになった。この部分では、fgh オントロジーにはキーワードとして lock しか含まれていないので、ブロックとして切り出されたと考えられる。
- 差分グラフ上で正の位置にある部分
  - 行番号 304 ~ 334

    文章中の単語で fgh オントロジー上に存在するキーワードは、backpack,bracket,external frame,front,rack であるが、オントロジーを利用

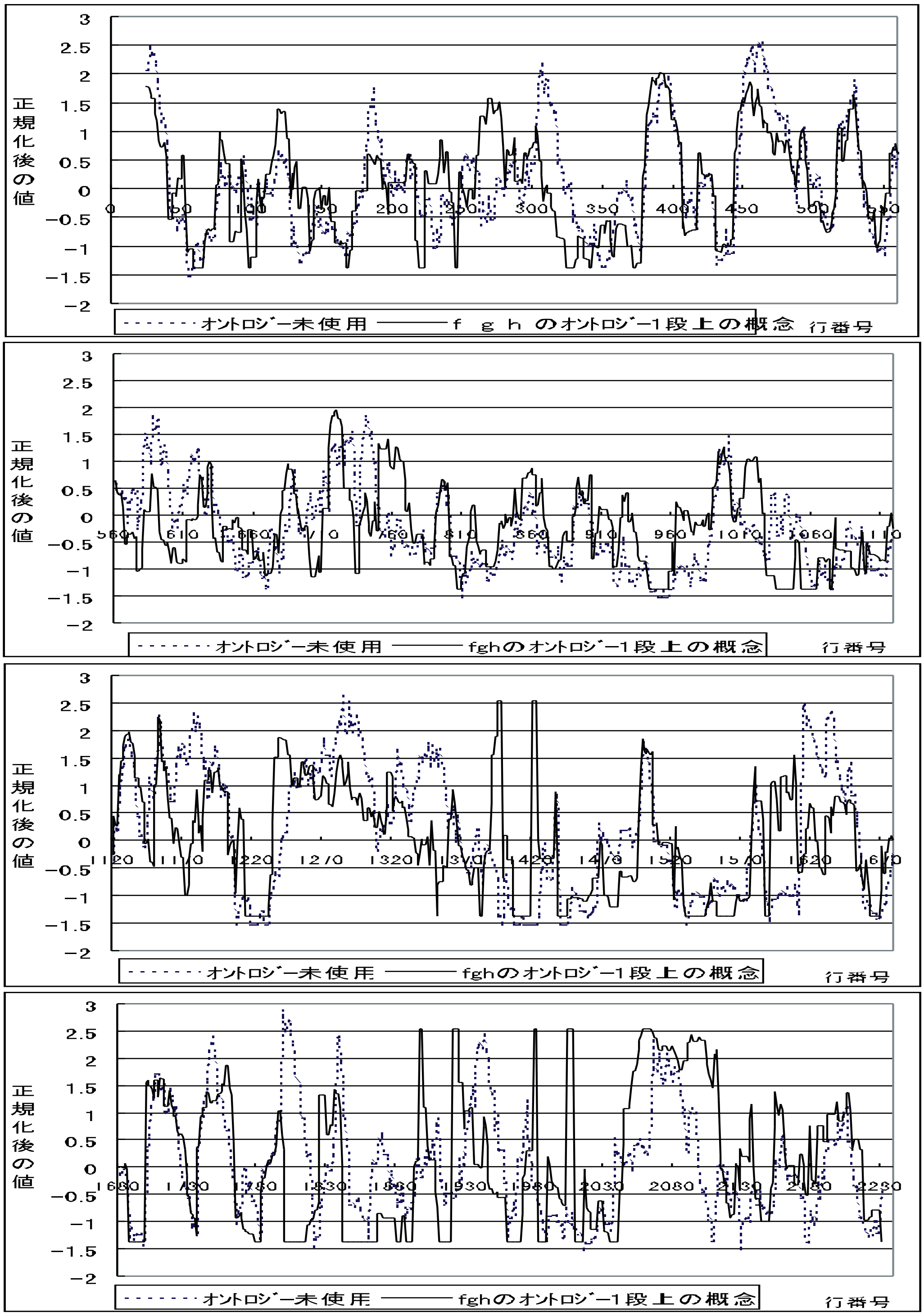

図 5.1 fgh オントロジー 1 段上の概念でまとめた時と、オントロジーを使わない時とを比較

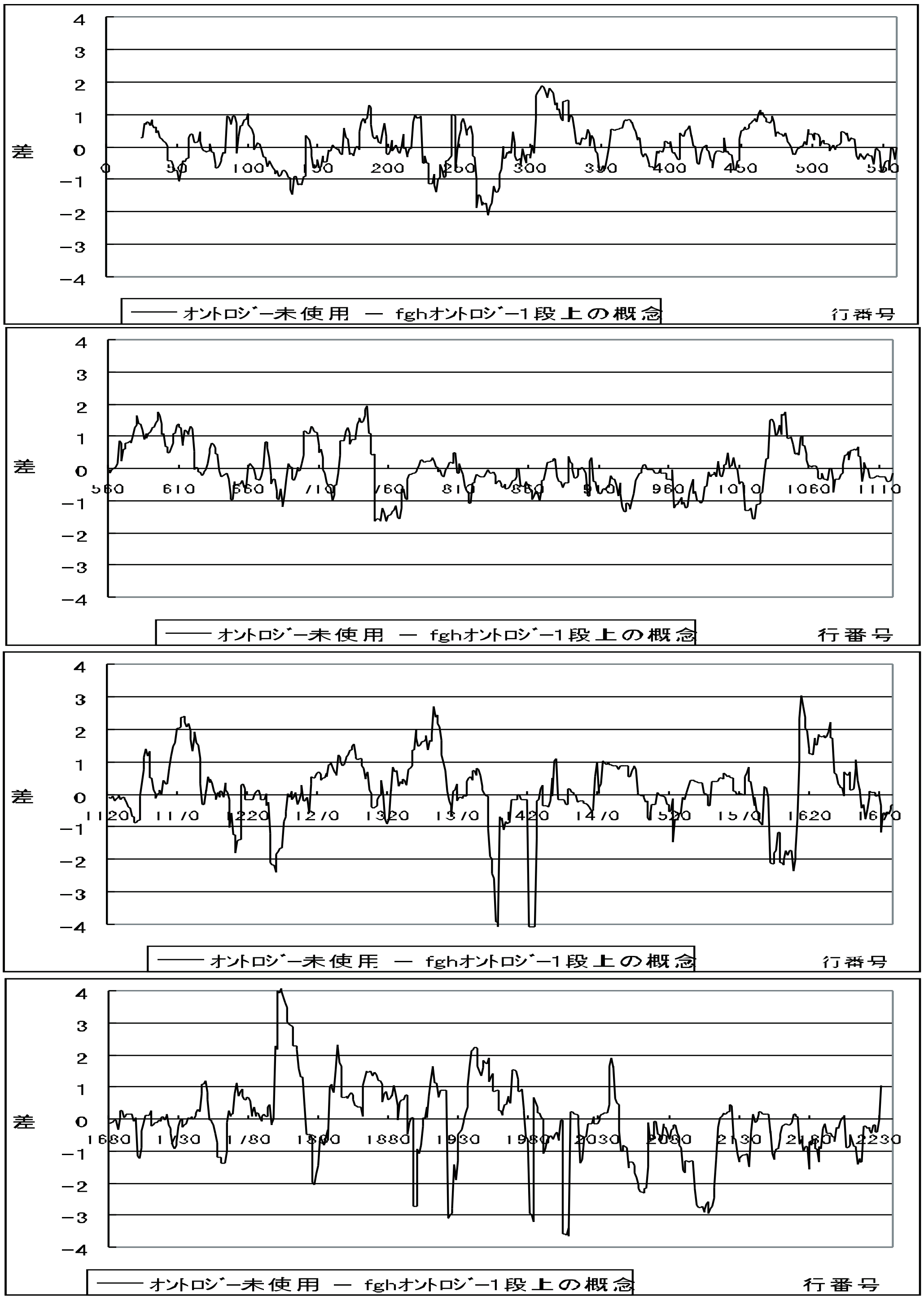

図 5.2 fgh オントロジー 1 段上の概念でまとめた時と、オントロジーを使わない時との差

して共通の概念にまとめられたものがなかった。この部分は、タイトル「チャイルドシートのようにハンドルバーへひっかける」である。

– 行番号 1165 ～ 1187

  fgh オントロジーの利用によって文章中の(キーワード 出現数)は、(backpackstrap 1),(strap 3) が概念 (strap 4) としてまとめられるが、fgh オントロジーに存在しない ijk キーワードが (tray 2),(shade 5)(net 5) のように出現するので、fgh オントロジーを使うと、ブロックを切り出しにくいと思われる。この部分は、タイトルは「トレーを使ってバックパックと自転車を結び付ける」の文章に位置する。

ijk と fgh オントロジーを利用した時とで、グラフの形状に違いが見られるところは、キーワードベクトルを構成しているキーワードの違いによるところが大きかった。

第 6 章

ブロック間の関係の発見手法の実験

6.1 ブロック間の関係を発見

前 2 章で述べたように構造化の第一段階では、キーワードベクトル法を用いることで、プロトコルデータを意味あるブロックに分割することが可能になった。

本章では構造化の第二段階として我々はブロック間に関係を発見する実験を行なう。ブロック間の関係として考えられるものの一つがブロックの内容の類似度合である。意味あるブロックとして切り出されたブロックの類似度合を求めることで、ijk データにおける内容が類似したブロックをつなぐことが可能になると思われる。それが ijk データの一つの構造化手法につながると考えられる。また、その構造化した結果を設計者に分かりやすく示すために、可視化を行なった。

類似した内容をもつブロックを発見する際には、ブロックの内容を捉える手法としてここでもキーワードベクトル法を用いることにする。この場合キーワードベクトルの使い方として、2 つのブロックのキーワードベクトルの角度が小さい方が、より類似した内容を表していると考える。

ijk データを構造化するために、切り出されたブロック全体に対して以下の手順で処理を進める。また、構造化するブロックとして手作業で選んだキーワードのみを使った手法 (オントロジーは使わない) を選んだ。

- 2 つ 1 組として、全組合せのキーワードベクトルを計算する
- 類似度が高いブロックの組合せから順にソートする
- 可視化のためにグラフとして表示させる

表 6.1 ブロック間の関係

ブロック番号	ブロック番号	関係の深さ
37	46	0.833
47	52	0.809
4	16	0.666
26	40	0.615
24	40	0.611
39	40	0.600
37	49	0.592
24	27	0.592
52	59	0.561
24	26	0.552
25	60	0.535
46	49	0.518
52	60	0.516
4	45	0.514
17	26	0.510
13	23	0.503
38	45	0.479
47	60	0.478
47	59	0.475
0	15	0.467
2	16	0.462
2	62	0.438
8	34	0.436
41	56	0.416
26	39	0.410
38	49	0.406
53	57	0.397

ソートした結果を表 6.1に示す。

この表から、ブロック 37 とブロック 47 の様に関係が深いブロックが多いブロックと、ブロック 3 やブロック 19 の様に関係の深いブロックがあまり存在しないブロックがあることに気付く。

意味あるブロック間で関係が深いと思われる個所を上位 20 組からいくつか取り出してそれぞれを分析する (図 6.1, 図 6.2)。

- ブロック 47, ブロック 52

  2 つのブロックに共通して出現するキーワード inch が両者を類似したブロックであると認識した。ブロック 47 は留めるために必要な部品のサイズ、について述べていて、ブロック 52 は何か支えるためのもののサイズ、について述べている。内容を見てみると直接のつながりはないと思われる。しかし、「もののサイズについての話題」という観点から見ると内容は類似しているといえる。2 つのブロックを人間が区切った文章と比較してみると、ブロック 47 はタイトル「tray の詳細 (サイズ) について」の文章の中間に位置している。ブロック 52 も「rear stay の高さについて」の文章の中間に位置している。

- ブロック 26, ブロック 40

  2 つのブロックに共通するキーワードは、release,seat,bike,rack,braze-on,frame などである。ブロック 40 はさらに screw,size も含んでいる。文章中のキーワードの数は、両者とも多い。ブロック 26 は、主にバックパックを留めるラックを真鍮で固定すること、について述べているがあまりフレームについては述べられていない。ブロック 40 は、ラックを真鍮で固定することについてより詳しく述べているのと同時にフレームのサイズについてもある程度の分量がある。この 2 つのブロックは、ある部分で話題が類似しているといえる。 2 つのブロックを人間が区切った文章と比較してみると、ブロック 26 はタイトル「強度と全体の重さを考慮」と「多機能のキックスタンドロック」に渡って切り出されていて、ブロック 40 は、「マウンテンバイクの設計図を見て構造を分析している」と「設計図を見ながら braze-on

1613 J centimetres or inches centimeters
1614 K centimetres oh inches
1615 J this is in inches oh wow so fifteen
1616 K mm mm
1617 J OK let's draw fifteen inches OK and it's probably what two inches deep (inaudible)
1618 I no we just need enough so that if the straps are in a funny position that it can still hold
1619 J what's this
1620 I that's this
1621 J is this a part that we can use
1622 K well this is one of the first things
1623 J oh it's the existing design
1624 I well that's the prototype
1625 J ahh that's expensive
1626 I yes
1627 J god fifty dollars huh (laugh)
1628 I OK here ah yeah lay this flat I want to just check up where you want to get to about here where's the paper
1629 K it's heavy
1630 I if you go too high on it you're gonna restrict people's ability to put stuff in it
1631 J yeah y'gotta get more than two inches oh but maybe
1632 I ahhh no you don't do you I mean you just have to catch the frame
1633 J oh yeah
1634 K it does curve some
1635 I it does there is room there for the straps
1636 J sales ( ) let's measure the high point of the frame then and go maybe a tad higher than that
1637 K you know what might be nice for em
1638 I I think you're two inches over the
1639 K it might be nice to have that curve built into our tray if we're injection moulding it in like
1640 I but what if someone wants to put like books or something else in there
1641 K that's true
1642 I yeah
1643 J just as long as that they they can't use it as a seat (laugh)
1644 I mm mm
1645 J thats why the tray would do it yeah
1646 I yeah that's what they would do yeah let's leave the back open although we don't want these straps coming out so let's put a em a shorter lip on the back or at least across the middle let these things the only question is how em or maybe there needs to be a ridge in there that you can put these inside of so that they just sort of stay up in there
1647 J I think that this needs to be additional anyway em cos for this to snap in
1648 I oh see that that's perfect see this thing well actually
1649 J yeah oh that could be the rear ridge
1650 I that could be the rear ridge right that comes up the side here
1651 J and it doesn't yeah and it just so that sets this dimension for us then
1652 I right so it comes up to here and across there
1653 J because we know what the width of that pack is across there er what's the width of the pack frame back there? packframe
1654 K eleven on the outside

図 6.1 ブロック 47 の本文

1786 J what

1787 I these may have to actually be up underneath because

1788 J well it's iffy 'cos then your pack would run under the seat wouldn't it

1789 I OK yeah you're right

1790 K you know what it might be

1791 J I think we're I think we're OK on there

1792 I (inaudible)

1793 J I think the thing that's kind of indeterminate is that this this length is gonna be

1794 I right

1795 J it might it might not it might not be as shown it might be in like this and if we had more time we could go in and figure it out

1796 K I think it's pretty close to fifteen point four

1797 I yeah

1798 J yeah

1799 I OK fifteen point four inches now you said the tool er I said eighteen inches

1800 K (inaudible)

1801 J well it'd be it would be less than fifteen point four right 'cos

1802 K mm mm

1803 J so this thing

1804 I what?

1805 J if you did this

1806 K it might be swung out enough to er get it kind of a triangle

1807 J oh oh oh oh 'cos you're saying a rear facing triangle

1808 K you don't also want to have to cantilever

1809 J if it went like it went back like this OK

1810 I it's probably going to because this thing is gonna be as long as the frame right

1811 J OK so you say fifty bucks

1812 I four inches roughly

1813 K yeah

1814 I it's gonna be about er (inaudible)

1815 K seventeen

1816 I inches so

1817 J so materials wise on this guy we've got thirty

1818 I plus

図 6.2 ブロック 52 の本文

についての議論」に渡って切り出されている。

- ブロック 37, ブロック 49

  2 つのブロックに共通するキーワードは、strap,bungee である。ブロック 37 は、bungee というキーワードが出現しているが、ほぼバックパックとラックを革ひもで留めることについての話題である。ブロック 49 中の話題は、革ひもの使い方、bungee について、など複数の内容になっている。革ひもの話題が存在しているとことから、内容的に類似しているブロックだと言える。2 つのブロックを人間が区切った文章と比較してみると、ブロック 37 は、タイトル「バックパックとラックを革ひもで留める」の文章の後半部分を切り出していて、ブロック 49 は「革ひもと、bungee を採用することにする」の前半部分を切り出している。

- ブロック 13, ブロック 23

  2 つのブロックに共通するキーワードは、backpack,front,back である。ブロック 13 の話題は、バックパックを後ろにのせる時の問題点について。ブロック 23 はバックパックを載せる位置を前後で比較している部分である。両者ともバックパックを載せる時に出てくる問題点についての話題なので内容的に類似しているといえる。2 つのブロックを人間が区切った文章と比較してみると、ブロック 13 は、タイトル「後ろに載せることを考えてバックパックの構造について議論」と少しずれた位置を切り出していて、ブロック 23 は「寝袋の存在に気付く」と「前部では、バランス、転回のための距離、ひざの位置が問題」に渡って切り出されている。

キーワードベクトルから類似性が高いと判断したブロックのいくつかを取り出して調べた結果、内容的に類似したブロックを取り出せていることがわかった。また、ブロック 40、ブロック 37 の様に一つのブロックに類似性の高いブロックが多く存在するものもいくつかあった。逆に、ブロック 11 の様にほとんど類似性のないブロックも存在した。プロトコルデータは何らか意味で類似性を持つ複数の話題と、他と類似性をあまり持たない話題から成り立っていると考えられる。

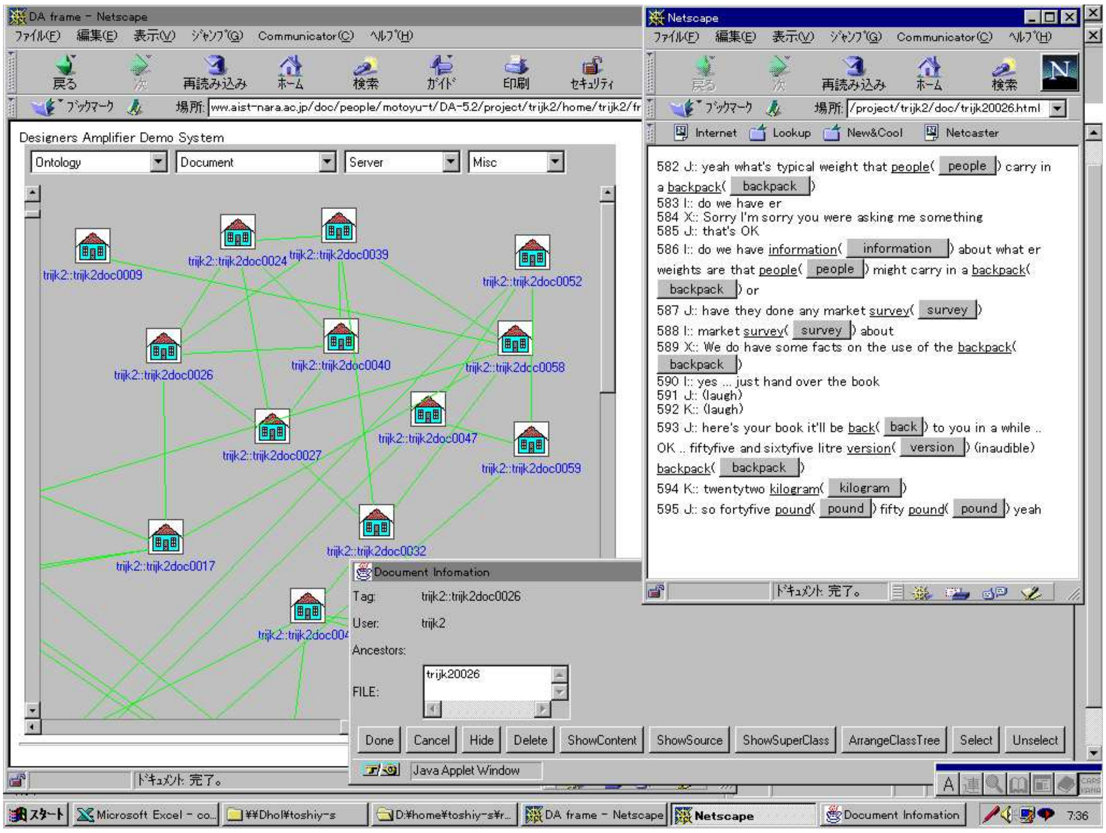

図 6.3 Designers Amplifier によるブロックとブロック間の関係の表示

6.2 ブロック間の関係を可視化

図 6.3は、Designers Amplifier システムにおけるブラウジング機能を利用して表示させたものである。このシステムを利用して、今回得られた意味あるブロックとブロック間の関係を表示させる。図中のアイコンが今回切り出したブロックの番号を示していて、アイコン間のリンクがブロック間の関係を示している。アイコンを選択すると、ブロック番号に該当する文章が画面上に表示されているウィンドウとして表示される。

第 7 章

考察

7.1 構造化した結果について

プロトコルデータを構造化してブロック間の関係を求めたものから、あるブロックに着目してそれに関係の深いブロックの組を取り出し、今回行なった実験結果について考察を行なう。

まず、ブロック 52 の様に、切り出されたブロックがタイトル「rear stay の高さについて」の内容を表す文章のの一部分である場合、そのブロックの文章だけでは明確な情報が得られない場合がある。このブロック内の文章のみでは、「何かを支えるためのサイズについて」までしか分からない。しかし、この「何か」に関しての情報は、ブロックを構成する部分のの数行前の文章中にある。従って、ブロックからは明確な情報を得られない場合、ブロックの前後の文章まで読めばある程度明確になることもある。特に切り出されるブロックが小さい場合、このようなブロックの拡大が必要だと思われる。

次に関係が深いと計算された、

- ブロック 38 タイトル「トレーを使ってバックパックと自転車を結び付けることについて提案」
- ブロック 43 タイトル「トレーの構造について」の一部
- ブロック 45 タイトル「トレーの構造について」の一部
- ブロック 47 タイトル「トレーの詳細について」の一部

を見る。

ブロックが位置する部分のタイトルを見るとトレーを利用することについて、トレー使用の提案からトレーの詳細な設計にいたるまでの一連の設計の流れが読みとれる。ブロックの内容を個別に見てみると、38ではトレーについて詳しく述べられているので、トレーに関する情報を得やすいが、他のブロックはプロトコルデータの性質上、そこだけ読むとはっきりとした情報は得にくい。しかしながら、関係が深いブロックとして38の存在が分かれば、38から得たトレーに関する情報をもとに他の3つのブロックについてより多くの前提知識を持って文章を読むことができる。また、可視化の段階でそれぞれのブロックに番号をつけているので、番号の若い順からブロックを眺めることによって、あるものに関しての設計の流れを捉えやすくなるといえる。

次に、

- ブロック47 文章タイトル「トレーの詳細について」の一部
- ブロック38 文章タイトル「トレーを使ってバックパックと自転車を結び付けることについて提案」
- ブロック54 文章タイトル「トレーのコストについて」
- ブロック57 文章タイトル「革ひものコストについて」
- ブロック37 文章タイトル「バックパックを革ひもでラックに固定する」の一部

を見る。

ブロック47では、トレーについての話題にのなかで革ひもについて触れている部分がある。そこで、革ひもの設計に関する情報を得ようと、ブロック37、57を読んでみると、それぞれ革ひもの提案と革ひものコストについて述べている。一方47と関係が深い38と54は、それぞれトレーの提案とトレーのコストについて述べている。ブロック番号の近い(プロトコルデータにおいて位置が近い)37と38はそれぞれ提案、54と57はそれぞれコストについて述べていることから、文章中に一つ大きなの設計の流れがあると考えられる。

設計の過程は、

- “問題提起”→“提案”→“展開”→“評価”→“決定”

という要素で成り立っていると考えられている [6]。本研究で用いたプロトコルデータにおいて、$37, 38 \rightarrow 47 \rightarrow 54, 57$ がちょうど設計の過程であるとされる “問題提起”,“提案”→“展開”→“評価”,“決定” に該当すると、読みとることができる。したがって、プロトコルデータ全体にも設計の過程が表されているということが分かる。

プロトコルデータ中の設計者の発話を分析した結果、設計者の発話つまり思考過程を表しているものは、設計過程のモデルそのものであることが分かった。

また今回の実験では、設計の際に重要だと考え手作業で選び出したキーワードの出現頻度をもとに、プロトコルデータを意味のあるブロックに分割し、そのブロック間に関係を発見した。その関係が、設計過程を表すことがあるとすると、それぞれの設計過程ではキーワードの出現が多くなるともいえる。

設計者は、プロトコルデータから過去の設計課題に対する設計過程の情報を得ることで、これから設計する対象物に関して考慮にいれなければいけない重要な点を理解することができると思われる。

7.2 キーワードベクトルを使ったブロック切り出しについて

ウィンドウサイズを設定してその中で出現するキーワード数から、キーワードベクトルを計算する方法では、そのブロックだけでは内容が薄いが関係の深いブロックを読むことでそれが意味あるものと理解できるブロックをある程度切り出せたと考えられる。

しかし、それでも切り出せなかったブロックがある。

その原因として考えられるのは、人間が会話の意味を理解して会話を区切る場合、あるキーワードが何回出現したかよりも、ただ一回出現するキーワードの意味を考えて区切る場合がある、ということである。例えば、「ラックの他の使い道を考える」という話題に対してキーワード rack より、使い方の話題に関してキーワード put,flip やキーワード以外の単語 here などが多数出現した場合である。

こういう問題に対して対処するためには、ある名詞に関わりの深い動詞の存在を考えなくてはいけない。例えば、rack には put が関係が深いのでそれぞれが出

現したら rack についての話題であると捉えるようにするなどである。

7.3 オントロジーの利用について

キーワードをオントロジーを使ってまとめてブロックに分割した結果、キーワードの出現の仕方によって間違って分割部分とされたところが、意味あるブロックとして捉えられる様になった。また、オントロジーを利用することによってブロックとして捉えられる部分の減少は少なく山の高さの低下にとどまった。逆に人間が細かく区切った文章を大まかな一つの意味を表すブロックとして捉えるようになった。

しかし、オントロジーを利用しても新たなブロックの切り出しができなかった部分が存在する。例えば、「cost を考えチューブとバックパックの素材について」の部分は、aluminum,weight,tube,backpack などのキーワードが存在するが出現回数が少ない上、3 段上のオントロジーではまとまらなかった。新たなブロックとして分割されなかった。また、上記の「ラックの他の使い道を考える」の部分でも、オントロジー上まとまらなかった。

今回オントロジーの利用は、全キーワードに対して 3 段上の概念でまとめるようにした。そうではなく、ある情報に対して詳しい話題のブロックが欲しいと考えた時、その情報に関しては、オントロジーを利用して何らかの概念でまとめ、その他はキーワードをそのまま利用して処理を行なうことも考えられる。また、tube と backpack を近いものとしてオントロジーを構築しておけば上記の例でブロックの切り出しは可能であったと思われる。

いずれにしても、オントロジーの利用の効果はその構造によって、決まると言える。

7.4 オントロジーの再利用性について

ijk データで fgh オントロジーを適用した実験結果より、以下のことがいえる。

ijk データでブロックとして捉えられなかった部分は fgh オントロジー（fgh のキーワードにより構成）に ijk のキーワードが含まれていないことがその原因と考えられる。したがって、同じ課題の与えられた設計におけるオントロジーの再

利用性は、その構造も重要であるが適用させる文章中に出現するキーワードをどれだけ含んでいるかのほうが影響が大きいといえる。

プロトコルデータ中のおよそ $375 \sim 550$ の範囲は、ブロックの形状が類似しているので ijk と fgh の両者に共通のキーワードが出現していると考えられる。それ以降は、両者のグラフに変化が現れている。文章内容をみると $375 \sim 550$ の範囲はバックパックと自転車の取りつけ方法として様々な提案を行なっているところである。それ以降はそれぞれの案に対しての詳細な設計である。同じ設計のテーマが与えられた時に、最初に提案される設計案は似ているということが分かる。そして、その後は設計するグループによって設計方法が異なってくるという一つの設計に関する情報を得ることができた。

7.5 出現頻度を基に半自動的にキーワード選出について

出現頻度を基に半自動的にキーワード選出した結果、全体的にブロック分割はうまく行かなかった。キーワードベクトルを利用する際に、設計にあまり関係のない単語によって、設計において重要だと思われていた単語の効果が減少するなど、好ましくない結果が出た。つまり、キーワードベクトル法を用いてプロトコルデータの内容を捉えるためには意味のあるキーワードを選択する必要があることが確かめられた。

第 8 章

結論と今後の課題

プロトコルデータのような構造化されていない文章を、その文章でキーワードと考えられるものを基にキーワードベクトルを用いて意味あるブロックに分割した。その結果得られたブロック間の類似性に着目して関連付けた。その構造化が、設計に関する議論の流れをつかむ一つの手法であることを示した。また、オントロジーを利用することによってキーワードのみでは切り出せなかった意味のあるブロックを切り出せたことを述べた。

また、プロトコルデータを構造化することによって、その中には設計過程のモデルとされるものをそのまま含んでいるということを確かめることができた。『自転車にバックパックを取りつけるための部品の設計』というテーマに対してもそれが当てはまることを示した。

以上より、今回提案したシステムは、設計における会話文のような議論に流れがある文章に対して有効であるといえる。

意味あるブロックに分割するには、キーワードの選択が重要であると述べた。しかし、キーワードの選択は時間と労力をかけて行なう必要がある。そこで、単語の情報量を計算することで自動的に重要な単語を抽出する手法 [5] を適応することなどが考えられる。

謝辞

主指導教官である西田豊明教授には、本研究を行う機会を与えて下さいました。また、研究を行うにあたっては、研究内容のみならず、研究態度や研究生活の面においても非常に有益な助言を数多くいただきました。ここに深く感謝いたします。

松本裕治教授には、お忙しい中、本研究の副指導教官となっていただきました。ここに深く感謝いたします。

武田英明助教授には、本研究の研究方針や研究内容などに関する有益な助言を数多くいただきました。また、常に研究が遅れがちな筆者に対して、常に叱咤激励して下さいました。ここに深く感謝いたします。

沢田篤史助手 (現在、京都大学計算機センター助教授)、上野敦志助手、久米出助手には、研究会などを通じて貴重な意見をいただきました。ここに深く感謝いたします。

知能情報処理学講座のみなさんには、研究生活を送るにあたり、様々な面からの協力をいただきましたことを感謝いたします。本講座博士後期課程の鷹合基行さんには、本研究の内容に関する有益な助言を数多くいただきましたことを感謝いたします。本講座博士後期課程の岩爪道昭さん、前田晴美さんには、論文作成において有益な助言をいただきましたことを感謝いたします。

小布施文代秘書には、事務手続きの面で協力して下さいましたことを感謝いたします。

参考文献

[1] G.Salton and M.J.McGill. *Introduction to Modern Information Retrieval.* MacGrew-Hill, 1983.

[2] Marti A. Hearst. Texttiling:segmenting text into multi-paragraph subtopic passages. In *Association for Computational Linguistics*, pp. 33–64, 1997.

[3] 鷹合基行, 武田英明, 西田豊明. 協調設計作業を実現するための設計者支援環境. 電子情報通信学会論文誌, 1998.

[4] Henri Christiaans Nigel Cross and Kees Dorst. *Analysing Design Activity.* 1995.

[5] 黒田直志, 大杉英一, 岩爪道昭, 武田英明, 西田豊明. ネットワークの情報を利用した概念獲得支援. 人工知能学会全国大会（第 11 回）論文誌, pp. 555–558, 1997.

[6] 武田英明, 冨山哲男, 吉川弘之. 知的 CAD の開発のための設計過程の分析と論理による形式化. 精密工学会誌, pp. 57(6):1047–1052, 1991.

[7] T.R.Gruber. Ontolingua: A mechanism to support portable ontologies. Technical Report 91-66, Knowledge Systems Laboratoty Stanford University, 1992.