特 別 研 究 報 告 題 目

単語間の概念距離を用いた クラスター分析による次元圧縮

Dimension Reduction by Cluster Analysis using Conceptual Distance between Words

指 導 教 員　　主査 出口 利憲　教授
副査 田島 孝治　准教授

岐阜工業高等専門学校 専攻科 先端融合開発専攻

2017Y26 服部 修平

平 成　31　年（2019　年 ）　　2　月　13　日 提 出

Abstract

We proposed the method to reduce the dimension using the concept distance between words for the similarity calculation between the documents. First, the concept distance between the words was calculated by using Japanese WordNet. Second, using cluster analysis of concept distances, the words were classified into the clusters by concept. The similarity between the documents was calculated using these clusters for dimension reduction. If the concept of the word is similar even if different words are used, the similarity includes the relation of the words.

In this study, the syllabuses of 51 college of National Institute of Technology (NIT) are used as the target documents. The reason is that knowledges to help learning may be got from the relations between the syllabuses. Furthermore, the information leading to the inter-college activity may be provided by finding the similarity between the departments or the colleges from syllabuses.

We checked the words in the clusters whose number is the same as the dimension used in the similarity calculation by LSI. As for the words in a cluster, the lowest superordinate concept of these words is an abstract one. The lowest superordinate concepts of the words classified in each cluster become concrete by increasing the number of the clusters. As a result, there was the case that the word clustering method was better than LSI by increasing the number of the clusters to classify words to some extent.

目 次

第1章　序論

1.1　序論

近年のコンピュータやスマートフォンの普及は著しく、ソーシャルネットワーキングサービスを利用することが一般的になりつつある。インターネット上で使用されるテキストは増加し続けているが、そのテキストには重要な情報から中身のない情報まで多くの情報が含まれている。その情報を人間が全て読み判断するには、テキストの量が膨大すぎて困難である。そこでコンピュータを使用してテキストから有用な情報を得るためのコンピュータ技術が開発された。これをテキストマイニングという。テキストマイニングの技術は膨大なテキストデータをデータとして処理することができるが、形態素解析や書き言葉と話し言葉の違いなど、自然言語から有用な情報を得るには課題も多く現状では完成された技術ではない。

本研究では、テキストマイニングの技術の一つである文書間の類似度計算について、新しく単語間の概念距離を利用した方法を提案する。また本研究手法を使用し、高専のシラバスから有用な情報を得ることを目的とする。平成 30 年度からシラバスが html で公開され Web スクレイピングの技術を使いテキストで入手できるようになったので、今回は 51 高専の全学科の全科目のシラバスを対象とする。

シラバス間の類似度を求めることで教科間の関係についての情報を得ることができる。教科間の関係からはいま学習している科目と近い科目がわかることで、次に学ぶ科目を意識して学習することができるようになる。高専間、学科間の関係から、他高専の学科との違いについてもわかるようになり、交流を持つ機会が得られるかもしれない。

シラバス間の類似度の計算方法としては、日本語 WordNet による概念距離を利用したクラスター分析による次元圧縮ができると考えた。しかし、この方法の結果が正しいか判断することは、指標がないため難しい。そこで従来から存在する、潜在的意味インデキシング (LSI) を用いた類似度計算法を比較対象として実験を行う。

これまでの研究 [1] から、本研究手法で使用するクラスター分析の方法で最も結果がよくなる方法はウォード法であると判明した。この結果は、樹形図から判断していたので、他の指標を用いて判断した方が良いと考えた。そこで、4 つのクラスター分析の方法で、クラスターに分類される単語と分類された単語の共通の synset を確認する実験を行った。結果として次元数を増加させることで、概念の近い単語に分類されることが分かった。そこで次元数を増加させることで、文書をよく分類できるようになると考え、次元数ごとの樹形図を比較する実験を行った。また、比較対象の LSI を利用した類似度計算では条件を合わせるために、日本語 WordNet に登録されていない単語を除いた場合の計算を行った。本研究手法では実験を行った結果から、単語をどの程度の概念に分類することで類似度がどう変化したかについて考察する。

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は、データベースの大量のデータから有用な知識を探し出す技術のことである。有用な知識とは、新規の有益な発見ことであり既存の知識は含まれない。データベースには大量のデータが存在する。この大量のデータに対し、人間による新しい有益な情報の発見には限界があり、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。そのために用いられる技術が、データマイニングである。

2.1.2　テキストマイニング

テキストマイニングとは、テキストデータを対象としたデータマイニングのことである。データの中にはテキスト形式で書かれたものも多くある。

キーワードによる検索は、既存の情報の発見に当たるのでテキストマイニングとはここでは別として扱う。この研究では、授業のシラバスの類似度計算を行い、単語間の概念距離を利用した類似度計算の方法とシラバスからの有用な知識の発見を目的とした。

2.1.3　形態素解析 [2]

形態素とは、意味の最小の単位である。形態素解析は、テキストを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。日本語は英語などのように、分かち書きとよばれる単語の区切りに空白を開ける記述をしない。単語が空白で区切られていないというのは、形態素解析をしづらくしている。形態素解析のツールは幾つかあるが、この研究では、MeCab というソフトを使用した。

2.1.4　MeCab

MeCab は日本語の書き方にも対応しており、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。MeCab で例として以下の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

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

自然言語とは、人工言語とコンピュータ言語以外の言語のことである。また、コンピュータ言語以外の言語としても自然言語は使われる。自然言語とコンピュータ言語の違いは、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。自然言語では曖昧な表現をしても、相手が解釈することで正しく伝わる。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。これは、命令に対する動作が複数あると常に同じ処理ができなくなるからである。

2.2.2　自然言語の曖昧さ

自然言語の曖昧さに多義性と類義性の二つがある。多義性は同じ単語でも複数の意味があり解釈が複数になること。類義性は異なる単語が同じ意味を示す場合のこと。例えば、おさめるには複数の漢字がありそれぞれ意味が異なる、雷と稲妻は同じ意味を持つ。自然言語の曖昧さは、コンピュータで自然言語を処理することを難しくしている。

第3章　研究に使用した技術・手法

3.1　計算手法

3.1.1　TfIdf[2]

TfIdf とは、文書中の単語の重みである。TfIdf は、文書中の単語の頻度を表す Tf (Term Frequency) と単語の情報量を表す Idf (Inverse Document Frequency) の積で求められる。文書が N 個のとき、各文書を d_i $(i = 1, 2, \ldots, N)$、文書 d_i のにおける単語が 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 は次の式で表される。

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

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

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

他に、出現頻度を総単語数で割ったものを Tf とする方法がある。出現頻度は長文であればあるほど増加するので、文の長さの違いが重要度として関係しないとする場合に用いられる。Idf は 1 を足すことによって重要度を 0 にならないようにする場合があるが、この研究では、すべてのシラバスに含まれる単語は類似度に影響しないと考えたので、すべてのシラバスに含まれる単語は重要度が 0 になるように式 (3.3) を用いた。他にも Idf の底が違う場合がある。

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。例えば、ベクトル $\vec{a}$ とベクトル $\vec{b}$ の cos 類似度は次の式で求められる。

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

3.1.3　主成分分析 [3]

実験や調査においては、多数の項目を記録することが多い。項目数が少ない時に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、結果の分析が難しくなる。これを解決する手法として、Hotelling 氏が

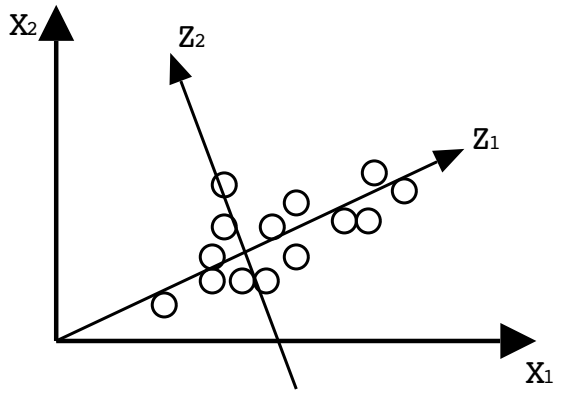


Figure 3.1: PCA in two-dimensional data

1933 年に提唱した主成分分析（Principal Component Analysis;PCA）がある。主成分分析は各データを独立に扱うのではなく、主成分と呼ばれる総合的な指標によってデータの持つ関係や特徴を表す。 これをもう少し詳しく説明すると、データが本来もっている情報の損失を最小限に抑えながら、 このデータを合成変数（主成分）に縮約して低次元化を行うことで、 多量のデータに埋もれた情報を把握するというのがこの手法である。 このことを、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、次式のような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り、それにはデータの分散に着目する。この例を示す為に Figure 3.1 のような 2 次元のデータを考える。この図において、データのばらつきが最も大きくなる方向に着目すると、z_1 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、これだけではデータの持つ情報を大まかに表したとは言えない。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 z_2 をとり、これによって情報量の損失を最小にしながら、x_1, x_2 から得られる特性を上手く把握することが出来る。

ここで、もし全てのデータが一直線上に並んだならば第 2 主成分は 0 となり、z_2 はデータの分析に全く役に立たないことになる。よって、データのばらつきである分散が大きいほど、情報を多く含んでいると言えるのである。以上の例は 2 次元という簡単な例であった為に、主成分分析はあまり役に立たないが、高次元であるとその効果は顕著に現れる。結合係数の決定の仕方としては、特異値分解に基づくものとスペクトル分解に基づくもの

がある。 第 i 主成分の結合係数 a_i は、前者においてはデータ行列 X の i 番目に大きな特異値 σ_i に対応する右特異ベクトルとして与えられ、後者においては行列 X^TX の i 番目に大きな固有値 λ_i に対応する固有ベクトルとして与えられる。前者については 3.1.5 項でもう少し詳しく説明することにして、ここでは後者について取り扱う。

ここでは分かりやすく第 1 主成分を取り上げて、結合係数が固有ベクトルで表されることを示す。 まず、第 1 主成分の分散 $\sigma_{z_1}^2$ は式 (3.7) のように与えられ、ここで t_1 は第 1 主成分得点と言い、これは n 番目のデータに対応する第 1 主成分の値をベクトルにまとめたものである。また V は分散共分散行列を表す。ここで、$N-1$ で割っていることから標準分散ではなく不偏分散を用いていることが分かり、後述する R の関数もこちらを採っているようである。不偏分散は、母集団が大きく標本が少ない時に向く。

$$\begin{aligned}
\sigma_{z_1}^2 &= \frac{1}{N-1} t_1^T t_1 \\
&= \frac{1}{N-1} (Xa_1)^T (Xa_1) \\
&= a_1^T (\frac{1}{N-1} X^T X) a_1 \\
&= a_1^T V a_1 \qquad (3.7)
\end{aligned}$$

第 1 主成分を最大にするには、この式 (3.7):f を式 (3.6):g の下で最大となるようにすれば良く、それにはラグランジュの乗数法から式を導き、それを結合係数 a_1 で偏微分して 0 とおけば良い。

$$\begin{aligned}
J_1 &= f + \lambda g \\
&= a_1^T V a_1 - \lambda (a_1^T a_1 - 1) \\
\frac{\partial J_1}{\partial a_1} &= 2Va_1 - 2\lambda a_1 = 0 \\
(V - \lambda I) a_1 &= 0 \qquad (3.8) \\
|V - \lambda I| &= 0 \qquad (3.9)
\end{aligned}$$

ここで I は単位行列を示し、式 (3.9) のように固有方程式が得られた。このことより、結合係数 a_1 は分散共分散行列 V の固有値 λ および固有ベクトルとして与えられることが分かる。この固有値が大きい主成分ほど情報を多くもっていることになり、大きい固有値から順に、その対応する主成分が第 1 主成分, 第 2 主成分, ... , 第 N 主成分に当たる。 また式 (3.7) に式 (3.8), 式 (3.6) を代入することにより、以下の式が導かれる。

$$\begin{aligned}
\sigma_{z_1}^2 &= a_1^T V a_1 \\
&= a_1^T \lambda a_1 \\
&= \lambda \qquad (3.10)
\end{aligned}$$

これは先程述べたように分散が情報の大きさを決定していることを示しており、最大値をとるべき $\sigma_{z_1}^2$ は最大固有値に等しい必要がある。これまでのことを以下第 N 主成分まで同様に導くことが出来る。以上のことより、結合係数 a は最大固有値に対する固有ベクトルとして求められ、これがスペクトル分解に基づく結合係数の決め方になる。 なお、ここま

で分散共分散行列を使って主成分分析を行う方法を記したが、相関係数行列を使って行う方法もある。どちらも一長一短であり、どちらが良いとは一概には言えない。

ここまで単に式を追ってきただけであったが、実際に測定や調査を行う時には、各項目は異なる単位系となることが多い。よって、単位の取り方により異なる主成分が得られることになり、同じ単位系であっても分散が大きく異なる項目に対して主成分分析を行えば、大きい方の影響を強く受けることになり、正しい結果が得られなくなる。そこで全ての項目を何らかの手法を用いて標準化する必要が出てくるが、広く利用されている方法として、各項目において平均 1・分散 1 となるように正規化するものがある。このような処置を施すことで、得られる結果の信頼性を高めることが出来る。

3.1.4 主成分の選択

主成分分析を行うことにより主成分をデータとして表すことができることは述べたが、主成分の数を決めることは重要な問題である。少なすぎれば情報の損失が多くなるし、多すぎては次元圧縮にならない。主成分の選択方法としては以下のようなものがある。

1. 固有値が 1 を越える主成分を採用する。
2. 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。
3. 累積寄与率がある値に達するまでの主成分を採用する。

これらをもう少し詳しく説明すると、1. は先述したように平均 1・分散 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。2. は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。3. 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%に達するまでの主成分数を採用する。累積寄与率（cumulative contribution ratio）は寄与率（contribution ratio）に関係するものであり、寄与率は次式で表される。

$$P_n = \frac{\lambda_n}{\sum_{p=1}^{P} \lambda_p} \tag{3.11}$$

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。一方、累積寄与率は次に示すように第 n 成分までの寄与率の総和で表される。

$$C_n = \sum_{i=1}^{n} P_i \tag{3.12}$$

3.1.5 潜在的意味インデキシング

潜在的意味解析（LSA：latent semantic analysis）は、文書行列を圧縮することで、分類を効果的に行う技法である。ここでいう文書行列とは式 (3.13) で表すような、重要度と

文書の行列である。

$$TD = \left(\begin{array}{c|cccc} Term & doc_1 & doc_2 & \cdots & doc_N \\ \hline w_1 & I_{w_1,doc_1} & I_{w_1,doc_2} & \cdots & I_{w_1,doc_N} \\ w_2 & I_{w_2,doc_1} & I_{w_2,doc_2} & \cdots & I_{w_2,doc_N} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ w_M & I_{w_M,doc_1} & I_{w_M,doc_2} & \cdots & I_{w_M,doc_N} \end{array}\right) \tag{3.13}$$

ここで、doc, w はそれぞれ N 個の文書、M 個の重要語を示し、I_{w_1,doc_1} は doc_1 における w_1 の重要度を表す。

このような文書行列は高次元であるため、分類や検索などの処理を行うには相当量の計算が必要になるのに加え、次元が増えるにつれて分類の妨げになる単語も増え、これがノイズのように邪魔になることがある。潜在的意味解析は、高次元の文書行列を低次元で近似的に表現する技術である。以下に、この LSA の計算法について簡単に説明する。

ある文書行列 TD に、次の行列式で表される分解（特異値分解) を行う。

$$TD = U\Sigma V^T \tag{3.14}$$

この式で U, Σ, V^T は行列を表す記号であり、右辺は三つの行列の積を表している。U は左特異（ターム）ベクトル、Σ は特異値を含むベクトル、V^T は右特異（文書）ベクトルと呼ばれる。この分解によって出た左特異ベクトルの列ベクトルは左にあるものほど重要度が高いので、左から最初の k 個だけで表される行列を U_k とする。すると、3.15 の行列積を求めることで、もとの文書行列に近似した行列を作成することができる。

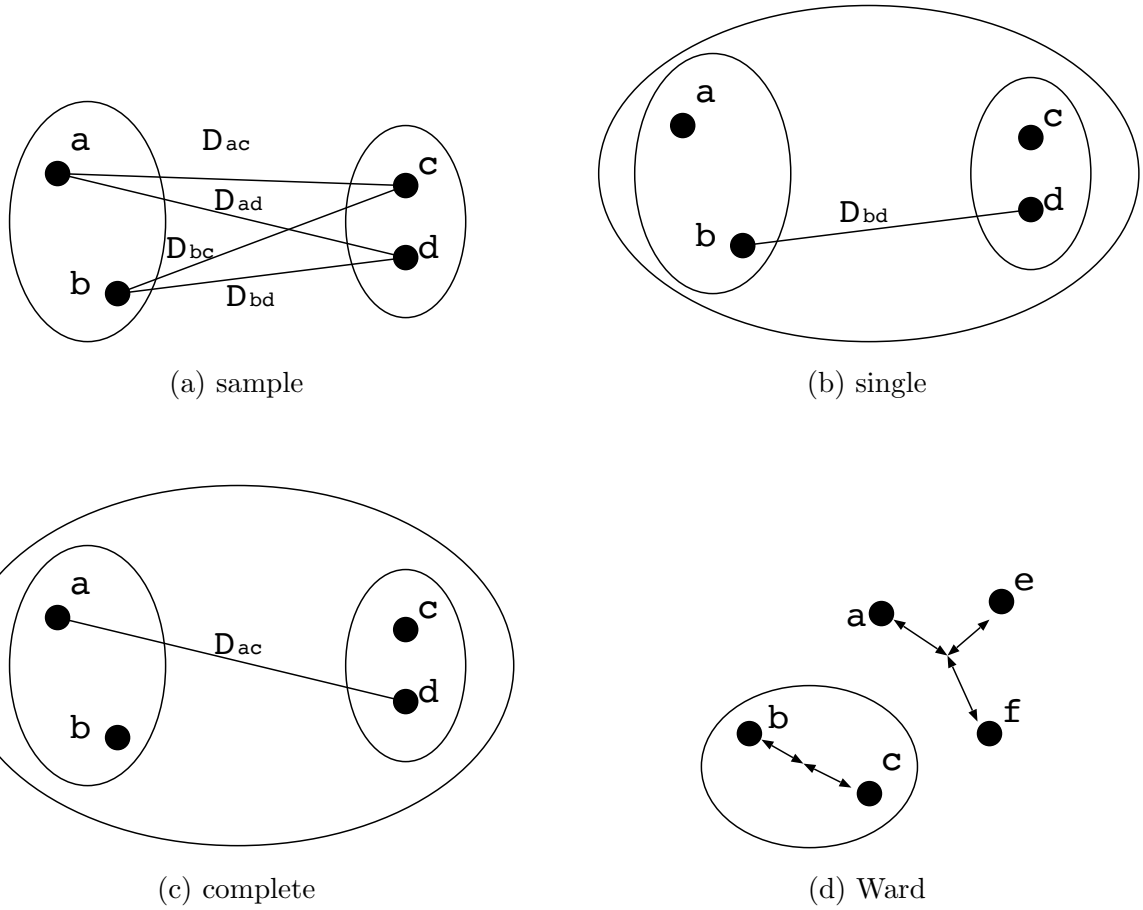
$$TD_k = U_k^T TD \tag{3.15}$$

3.1.6 クラスター分析

クラスター分析とは、与えられたデータをいくつかの集合に分類するデータ解析手法のことである。分類された後の集合をクラスターと呼ぶ。クラスター分析には、分類が階層的になる階層的クラスター分析と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手法を使用するので階層的クラスター分析について説明する。階層的クラスター分析とは、データ間の類似度または非類似度に基づいて、最も似ているデータから順次集めてクラスターを形成していく。R 言語ではクラスターを形成していく様子を樹形図で示すことができる。階層的クラスター分析はクラスター間の距離を決める方法にいくつかの種類があるが、その中から最近隣法、最遠隣法、群平均法、ウォード法について説明する。

最近隣法 2 つのクラスターの中から、最も近いデータ間の距離を 2 つのクラスターの距離とする方法。Figure 3.2(a) では、クラスター ab とクラスター bc の距離として距離 D_{bd} を選択し、Figure 3.2(b) のクラスターができる。

最遠隣法 2 つのクラスターの中から、最も遠いデータ間の距離を 2 つのクラスターの距離とする方法。Figure 3.2 では、クラスター ab とクラスター bc の距離として距離 D_{ac} を選択し、Figure 3.2 のクラスターができる。



(a) sample (b) single

(c) complete (d) Ward

Figure 3.2: Cluster

群平均法 2 つのクラスターの中から、それぞれデータを一つずつ選び距離を求め、それらの距離の平均値を 2 つのクラスターの距離とする方法。Figure 3.2(a) では、クラスター ab とクラスター bc の距離として、距離 $D_{ac}, D_{ad}, D_{bc}, D_{bd}$ の平均を計算し、新しいクラスターを形成する。

ウォード法 2つのクラスターを融合した際に、群内の分散と群間の分散の比を最大化する基準でクラスターを形成していく方法。Figure 3.2(d) の場合、データ b、c からなるクラスターが形成される。

最近隣法と最遠隣法にはそれぞれチェーンと拡散現象という性質があるので、次にその説明を示す。

チェーン クラスターが大きくなるにつれ、他のデータと最短距離を多く持つようになり、次のクラスターの形成の候補に選ばれやすくなる現象。

拡散現象 クラスターが大きくなるにつれ、他のデータと最長距離を多く持つようになり、次のクラスターの形成の候補に選ばれにくくなる現象。

3.2 Web スクレイピング

Web スクレイピングとは、ウェブサイトから情報を入手するコンピューターソフトウェア技術である。アクセスできる情報を効率的に収集することができる技術であるが、アクセス数が多くなると不正アクセスとみなされる場合や web サイトによっては利用規約で禁止されている場合があるので注意する必要がある。

3.2.1 研究での Web スクレイピング

これまでの研究 [1] では平成 28 年度の岐阜高専の電気情報工学科の 58 科目のシラバスを対象としてきた。しかし、対象のテキストが一つのフォルダしかないというのは研究を行う上で問題の一つだった。解決策として対象のシラバスを増やすことを考えたが、シラバスは各高専がそれぞれ PDF で Web 上に載せられており全てをテキストに変換して入手するのは難しい作業であった。しかし平成 30 年度から、Web シラバス (https://syllabus.kosen-k.go.jp/Pages/PublicSchools) に全高専のシラバスが html 形式で載せられるようになったため、全高専の全学科の全教科のシラバスを Web スクレイピングを使用し入手することにした。平成 30 年度に登録されている全高専の全学科の全教科のシラバスをを対象に、R 言語で Web スクレイピングを行なった。

51 高専の 477 学科で約 3 万の科目のシラバスを入手することができた。

3.2.2 Web シラバスの問題

Web シラバスには、平成 30 年度の今年から始まったためかいくつか問題があった。仙台高専のように数科目しか登録されていない学科や岐阜高専の人文教育、自然教育のように学科として存在するものの、今年度は一科目も登録されていない学科などがあった。他にも、岐阜高専の電子システム工学専攻と建設工学専攻ような既に存在しないため、今年度は一科目も登録されていない学科や久留米高専の材料工学科のように名前を変更した学科があり、学年で学科名の違う二つ学科が存在する高専などが存在した。一番影響が大きいと考えられるのは、一般科目と専攻科目を合わせて学科に登録している高専と一般科目を分けて登録している高専が存在することだった。

3.3 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る。多くの関数が用意されているため、複雑な計算を数行で実行できるという特徴がある。また、オープンソースかつフリーのソフトウェアである。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。この実験では、式 (3.13) のような、行列を主に使用するのでこの R 言語を主に使用することとした。R 言語では、Web スクレイピング、TfIdf、cos 類似度、主成分分析、LSI、クラスター分析はそ

れぞれ関数が用意されているので非常に簡単に実行することができる。以下にそれぞれの関数について書く。

3.3.1 Web スクレイピング

最初にパッケージ lubridate, pathological, rvest, stringer, tidyverse,XML を導入する。rvest に含まれる read_html 関数を使用することで、シラバスの html を読み込むことができる。読み込んだ html のデータのテキスト部分を rvest に含まれる html_node 関数で Xpath を指定し、入手した情報をテキスト形式で保存した。

3.3.2 TfIdf

パッケージ RMeCab に含まれる docMatrix 関数を使用することにより、フォルダのディレクトリと引数 pos に “Tf”、引数 weigth に “名詞” を指定すれば、フォルダ内のファイルを読み込み、式 (3.13) の形で名詞のみの Tf を計算することができるようになる。また、“TfIdf” を引数 weight で指定することもできるが、Idf は 1 を足す方法となっている。このため、本研究では Idf を別で計算し、Tf にかけることにより TfIdf を求めた。

3.3.3 cos 類似度

パッケージ proxy を読み込むことで、デフォルトで存在する dist 関数の引数に “cosine” が選択できるようになり、dist 関数にデータと引数 method に “cosine” を指定することで cos 類似度が計算できるようになる。

3.3.4 主成分分析

prcomp 関数に行列を引数として渡し、 scal を”TURE”に設定することで相関係数行列がもとめられる。デフォルトは FALSE で分散共分散行列で主成分分析が行われる。また、summary 関数を使用することで、標準偏差、寄与率、累積寄与率が求められる。

3.3.5 LSI

svd 関数を使用することで、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 次元までの圧縮ができる。

3.3.6 クラスター分析

hclust 関数を使用することでクラスター分析ができる。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。また、ctree 関数に hclust 関数の結果を渡すことで、単語が何番目のクラスターに属するかが分かる。

3.4 日本語 WordNet[4]

日本語 WordNet とは、日本語の概念辞書である。個々の概念が synset という単位にまとめられており、単語は一つ以上の synset に所属している。また、他の synset と意味的に結びついている。日本語 WordNet は英語 WordNet をもとに構築されている。この研究では、単語間の概念距離を求めるために使用した。

3.4.1 日本語 WordNet の問題

日本語 WordNet を利用することによる問題があることがわかっている。日本語 WordNet には約 10 万の単語が登録されているが、文書中の名詞が全て辞書に登録されているわけではない。辞書に登録されていない単語の概念距離は求められない。それゆえ、概念距離を求められる単語は文書に実際に含まれている全ての名詞より少なくなるという問題がある。

3.4.2 日本語 WordNet による単語間概念距離

日本語 WordNet の synset にはルート synset から上位下位の synset が木構造のように存在し、概念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、類似度計算に使用する。二つの単語の synset の関係により、概念距離を求める式が異なるので、以下にその方法を示す。

同じ synset に存在する場合 概念距離は 0 とする。

違う synset に存在する場合 概念距離は式 (3.17) を使用して求める。

違う synset に存在し、synset 間に複数のルートが存在する場合 全てのルートの中でC_{ab}が最大になるルートに対し式 (3.17) を使用して求め、その中から最小値を概念距離とする

複数の synset に属する場合 それぞれ式 (3.17) を使用して求め、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 と B があるとき、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。[5]

$$S_{a,b} = \frac{2C_{ab}}{L_a + L_b} \tag{3.16}$$

式 (3.16) は最大値が 1 になるよう正規化されているので、式 (3.17) で類似度 $S_{a,b}$ は概念距離 $D_{a,b}$ に変換できる。

$$D_{a,b} = 1 - S_{a,b} \tag{3.17}$$

3.4.3 クラスター分析による次元圧縮

クラスター分析は次元圧縮の方法ではない。しかし、概念距離を使用して名詞のクラスター分析を行うことより、名詞をいくつかのクラスターに分類することはできる。ここで、概念距離が近いものは、類似度が高いものである。これを利用し、分類した名詞をクラスターごとにまとめることを次元圧縮として行った。この次元圧縮の方法を、名詞のクラスター分析による次元圧縮とする。

TfIdf を重要度とした式 (3.13) のような文書行列に、名詞のクラスター分析による次元圧縮を行うことで、クラスターと文書からなるクラスター-文書行列が作成される。このとき、クラスター-文書行列の重要度は、式 (3.18) で表されるクラスターと名詞の行列と元の文書行列との積で求められる。

$$CW = \left(\begin{array}{c|cccc} Term & w_1 & w_2 & \cdots & w_M \\ \hline C_1 & a_{1,1} & a_{1,2} & \cdots & a_{1,M} \\ C_2 & a_{2,1} & a_{2,2} & \cdots & a_{2,M} \\ \vdots & \vdots & \vdots & \ddots & \vdots \\ C_N & a_{N,1} & a_{N,2} & \cdots & a_{N,M} \end{array}\right) \tag{3.18}$$

各クラスターを $C_i(i = 1, 2, \ldots, N)$、単語を $w_j(j = 1, 2, \ldots, M)$ とする。$a_{i,j}$ は次の式で与えられる。

$$a_{i,j} = \begin{cases} \dfrac{1}{|C_i|} \displaystyle\sum_{k \in C_i} S_{k,j} & (j \in C_i) \\ 0 & (j \notin C_i) \end{cases} \tag{3.19}$$

第4章　実験方法と準備

4.1　類似度計算の方法

実験で使用する文書間の類似度計算の方法として、LSI を利用した類似度計算の方法と単語間の概念距離を利用したクラスター分析による次元圧縮 (以下、本手法) を使用する。以下に、それぞれの計算方法について述べる。

4.1.1　TfIdf

R 言語で TfIdf を求める。方法は 3.3.2 節で説明した通りで、docMatrix 関数にファイル名、引数 pos に “名詞”、method に “Tf” を指定すると自動で R 言語が Tf を計算する。計算された Tf は式 (3.13) の形式の行列で表示され、doc, w はそれぞれ N 個の文書、M 個の名詞を示し、I_{w_1,doc_1} は doc_1 における w_1 の Tf を表す。Idf は式 (3.3) を R 言語で計算し、Tf の行列にかけることで TfIdf の行列を求めた。

4.1.2　LSI による類似度計算

R 言語を使用し、4.1.1 節で求めた TfIdf の行列を 3.3.5 節で述べたように svd 関数に引数として渡すことで左特異ベクトルが得られる。主成分から次元圧縮する次元数 k を決定する。次元数 k は 3.3.4 節で説明した prcomp 関数と summary 関数を用いて、累積寄与率が 80%を超える次元を次元数 k とする。左特異ベクトルの k 列目までの転置行列と TfIdf の行列を掛け合わせ、次元削減した行列を求める。次元削減した行列を cos 類似度を利用し文書間の類似度を求める。

4.1.3　単語間の概念距離を利用したクラスター分析による類似度計算

R 言語で docMatrix 関数を使用し、対象とする文書から名詞のみを取り出し、csv 形式に保存する。名詞間の概念距離を、C 言語で 3.4.2 節で説明した方法で求め csv 形式で保存する。名詞間の概念距離の csv ファイルを R 言語で読み込み、dist 関数で距離行列にする。距離行列となった名詞間の概念距離を式 (3.18) のクラスター-単語行列に変換する。4.1.1 節で求めた TfIdf の単語-文書行列に掛け合わせることで、クラスター-文書行列を求める。クラスター-文書行列が単語-文書行列を次元圧縮した行列となる。

求めたクラスター-文書行列を cos 類似度を利用して文書間の類似度を求め、hclust 関数を利用しウォード法でクラスター分析し、plot 関数で表示することで樹形図を求める。

4.2 準備

実験の前段階として以下の作業を行った。

- R、MeCab、日本語 WordNet のインストール
- R に 3.3 節で述べた全てのパッケージのインストール
- シラバスを Web スクレイピングでテキストとして取得する
- 高専と学科のテキストを作成する

4.2.1 シラバスを Web スクレイピングでテキストとして取得する

R 言語で Web スクレイピングするためのプログラムを作成した。Web シラバスのウェブサイト構成は、ホーム、各高専のページ、各学科のページとカリキュラムマップ、各学科のページから各教科のシラバスのページとなっている。各学科のページとカリキュラムマップは年度が指定でき過去の年度も確認できるようになっているが、各高専から各学科のページにアクセスすると平成 30 年度になるよう設定されているようである。

シラバスのページから科目基礎情報、到達目標、ルーブリック、学科の到達項目との関係、教育方法、授業計画、評価割合の 7 つの項目の内容を Xpath を指定することで取得し、テキストとして保存した。学科の到達項目との関係は何も書かれていない場合が多い。約 3 万の科目のシラバスを入手することができた。シラバスの文字コードは UTF-8-unix とした。

4.2.2 高専と学科のテキストを作成する

学科のテキストは、その学科のページに登録されていたシラバスを全て結合し、学科のテキストを作成した。高専のテキストは、その高専にある学科のテキストを全て結合し作成した。

4.2.3 作成したテキストの問題

作成したテキストには作成した時点で 3.2.2 で述べた問題を含め、いくつか問題があることがわかっているのでここで述べる。また、いくつか確認したテキストは修正をしたが、約 3 万のシラバスのテキストを全て確認して修正する作業は行なっていない。

- 登録されている科目が少なくても、一つの学科として扱っている
- 一般学科は実際に生徒の所属している学科名ではないが一つの学科として扱っている
- 高専ごとに、一般科目が一般学科として分かれているか、それぞれの専門学科に登録されているかが違う

- 後者の場合、それぞれの学科で同名の一般科目なら内容はあまり変わらない、よって同じ高専の学科間では一般科目による類似度が出ると思われる。高専のテキストを作成した場合、一般科目の内容が重複する。
- 一文でも、web シラバスで文の途中に改行コードがあり、単語が切られる場合がある。
- 計算に影響はないが、不自然に大量の改行やタブが入っている場合がある
- 改行かタブ、スペースが必要な位置に入らない場合がある。

第5章　実験1:クラスターの詳細の確認

これまでの研究 [1] では、概念距離をクラスター分析する方法でウォード法を類似度計算に使用するとしたが、結果の判断は樹形図による文書の類似度からだった。そこで新たに、クラスターに分類されている単語の数やクラスターの概念について確認し、ウォード法で概念距離をクラスター分析するべきか判断することにした。対象とする文書は、これまでの研究で使用していたファイルを少し改良したファイルであり、平成 28 年度の岐阜高専の電気情報工学科で使われていたシラバス 58 科目で平成 30 年度のシラバスではない。この文書には 1518 語の名詞が含まれており、累積寄与率が 8 割を超えるのは 37 次元とわかっている。

5.1　確認の手順

- 文書に対し、形態素解析を行い名詞を取り出す。
- 名詞間の概念距離を日本語 WordNet で求める。990 の名詞が登録されていた。
- 名詞間の概念距離を、hclust 関数を使用し、四つの方法でそれぞれクラスター分析する。
- 結果を ctree 関数を使用することで、990 の名詞を 37 のクラスターに所属する名詞ごとに分類する。
- 分類した名詞ごとに全てに共通する上位概念の synset を日本語 WordNet で求める。共通した上位概念の synset がそのクラスターの synset となる。
- クラスター synset がルート synset からどれだけ離れているかを確認する。クラスターごとに分類された名詞を確認する

5.2　クラスターの詳細の確認

以下の Table 5.1, Table 5.2, Table 5.3, Table 5.4 に、最近隣法と最遠隣法、群平均法、ウォード法の四つの方法で 37 のクラスターに名詞を分割した結果を示す。

Table 5.1: The details of the cluster using single for cluster analysis

Number of stages from root synset	Total of cluster synsets	Total words
0	1	897
4	2	3
5	2	2
6	6	6
7	10	10
8	5	5
9	7	8
10	1	1
11	2	2
14	1	1

Table 5.2: The details of the cluster using complete for cluster analysis

Number of stages from root synset	Total of cluster synsets	Total words
2	8	207
3	12	424
4	11	309
5	3	47
7	2	2
14	1	1

Table 5.3: The details of the cluster using average for cluster analysis

Number of stages from root synset	Total of cluster synsets	Total words
1	5	332
2	8	389
3	8	143
4	5	107
5	2	8
6	3	5
7	2	2
8	2	2
12	1	1
14	1	1

Table 5.4: The details of the cluster using ward for cluster analysis

Number of stages from root synset	Total of cluster synsets	Total words
1	1	29
2	8	169
3	12	327
4	10	307
5	6	158

5.3 確認の結果

5.3.1 最近隣法

ルート synset である 0 段目にほぼ全ての単語が分類されている。他の synset は、1 つのクラスターに 1 つ 2 つしか単語が分類されていない。これはチェーン現象のためと考えられる。ルート synset は、単語間の概念が関係なくとも全ての単語の共通の synset なので、概念距離が関係なく単語をクラスターに分類する意味がない。よって、最近隣法は本手法では使用できないことがわかった。また、1 つのクラスターに 1 単語しか分類されていない場合は、式 (3.19) で $a_{i,j}$ は 1 になる。以下に 1 単語で 1 クラスターとなった 28 の単語を示す。

- 委員、定常、自動、母集団、新鮮味、難解、主体、能動、広範囲、林、生、マクスウェル、アンペア、コンクリート、一貫、劣化、イオン、日本語、三角、種々、ハッシュ、国際、メモ、コンピューターグラフィックス、電子、中学校、高度、日本

5.3.2 最遠隣法

ルート synset から 5 段目までにほぼ全ての単語が分類された。1 単語で 1 クラスターに分類されている単語を確認したところ、3 段目に「薄膜」、7 段目は「自動車」と「小川」で、14 段目が「日本語」となっていた。この 4 つの単語は単語自身が所属している synset とクラスター synset が同じだった。他のクラスター synset を確認したところ、30 単語前後のクラスターが 21 個あり、残りの 5 つのクラスターは 70 以上の単語が分類されており、合計で 499 単語が分類されていた。200 単語が分類されているクラスターが 1 つ存在した。多く単語が 5 つのクラスターに分類されていることが類似度計算に影響を与えると考えられるが、単語の分類はある程度できていると考えられる。

5.3.3 群平均法

全てのクラスター分析手法の中で一番多くの段数に分類されている。ルート synset から 1 段目に半分の単語、2 段目までに 3 分の 2 以上の単語が分類されている。クラスター synset を確認したところ最遠隣法でも見られたように、いくつかのクラスターに多くの単語が含ま

れており、6 のクラスターに 674 単語が分類されていた。ルート synset から 1 段目に半分の単語が集まっていることと 1 単語しか分類されていないクラスターが存在するのは、チェーン現象によるものと思われる。ルート synset から 1 段目は physical_entity,abstract_entity の 2 種類しかないため、1 段目に分類された単語の類似性はほぼなく単語をクラスターに分類しても類似度を求めるには適切でないと考えられる。よって、単語間の概念距離を求める意味があまりなく類似度計算を行うのは難しいと考えられる。

5.3.4 ウォード法

ルート synset から 5 段目までに全ての単語が分類された。クラスター synset を確認したところ、50 以上の単語が分類されているクラスターは 3 つ存在した。3 つクラスターに分類された単語の合計は 218 単語で、1 つは 100 単語が分類されていた。残りのクラスターには 30 前後の単語が分類されていた。他の方法の結果ような、1 単語しか分類されていないクラスターは存在しなかった。

5.3.5 結果

クラスター synset を確認したところ全ての結果で、ルート synset から 3 段目以内に半分以上の単語が分類されていることがわかった。四つのクラスター分析による単語の分類を調べたところ、群平均法と最近隣法ではチェーン現象によって一つのクラスターに多くの単語が集まっている。チェーン現象によって大きくなるクラスターの synset はルート synset から近くなっている。この二つの方法では、多くの単語が分類されるクラスターの synset がルート synset に近くなっているため、本手法で概念距離を求めるのは難しいと考えられる。最遠隣法とウォード法はルート synset から 2 段目から 4 段目までにほとんどの単語が分類されている。ウォード法の方は、ルート synset から 1 段目に 29 の単語が分類されているが、4 つの方法で一番多くの単語がルート synset から遠い synset に分類されている。最遠隣法の方は、5 つのクラスターに 499 単語が分類されている。また、ルート synset から遠いクラスター synset が存在するが、1 単語で 1 つのクラスターに分類されてるため (以下、1 単語 1 クラスター)、本手法による類似度計算の際には、その単語を含むかどうかで類似度が変わることになる。

5.3.6 考察

5.3.5 節の結果から、本手法ではウォード法か最遠隣法によって単語間の類似度を求めることにした。いくつかクラスター分析による単語の分類によって類似度計算に影響を与える問題がわかった。

クラスター synset を確認した結果、ルート synset に近い synset の概念は抽象的な概念が多いことがわかった。抽象的な概念にクラスター synset が分類されると、単語の synset が所属する単語自身の本質的な意味の概念よりも抽象的な概念に分類されることになり、類似性があまりない他の単語との概念距離を求めることになる。

Table 5.5: The details of the cluster using complete for cluster analysis in 100 dimensions

Number of stages from root synset	**Total of cluster synsets**	**Total words**
2	5	38
3	16	166
4	28	394
5	19	231
6	14	108
7	6	17
8	8	31
9	1	2
10	1	1
11	1	1
14	1	1

また、1 単語しか分類されていないクラスターが存在した。1 単語 1 クラスターの場合、式 (3.19) の $a_{i,j}$ は 1 になる。よって、そのクラスターに分類された 1 単語の TfIdf の値が大きい場合は大きく類似度が変化することになり、その単語を含むか含まないかによって類似度が変化することになる。

本手法では、単語間の概念距離を使用するので、単語自身の持つ概念からは抽象的すぎるルート synset に近い概念に分類されていることは望ましくないと考えた。そこで、次元数を増やすことにより 1 つのクラスターに分類される単語数が減り、クラスター synset はルート synset から遠くなっていくと考えた。しかし、1 単語 1 クラスターも増えていくと思われる。クラスター synset がルート synset から遠くなれば、クラスター synset は抽象的な概念から具体的な意味の概念になる。具体的な意味の概念に分類された単語は類似性が高くなる。単語の類似性が高くなると概念の近い単語によって類似度計算を行うことができる。圧縮する次元数を増やした場合のクラスター synest とクラスターに分類される単語を調べることとした。

5.4 クラスター数と単語の関係

本手法では、次元数を増加させる場合、概念距離を求められた単語数まで次元数を増加させることができるので最大 990 次元まで増やすことができる。単語数 = 次元数となった場合は、TfIdf を cos 類似度で類似度計算した場合と同じになるので本手法を行う意味はないので実際は 989 次元までである。LSI では文書数までしか次元数を増加させられない。

次元数を増やした場合のクラスターに分類される単語の変化を調べていくことにした。5.1 節の手順を分類するクラスター数を変えて行う。以下の Table 5.5, Table 5.6, Table 5.7, Table 5.8, 次元数を 37 から 100 と 200 に増やした場合の最遠隣法とウォード法によるクラスターに分類される単語の変化について示す。

Table 5.6: The details of the cluster using ward for cluster analysis in 100 dimensions

Number of stages from root synset	**Total of cluster synsets**	**Total words**
1	1	3
2	9	58
3	18	177
4	32	358
5	18	212
6	13	125
7	5	31
8	4	26

Table 5.7: The details of the cluster using complete for cluster analysis in 200 dimensions

Number of stages from root synset	**Total of cluster synsets**	**Total words**
2	1	5
3	17	70
4	40	281
5	39	226
6	45	241
7	26	95
8	15	49
9	8	10
10	5	9
11	2	2
12	1	1
14	1	1

Table 5.8: The details of the cluster using ward for cluster analysis in 200 dimensions

Number of stages from root synset	Total of cluster synsets	Total words
2	3	21
3	16	78
4	44	241
5	39	239
6	45	208
7	27	129
8	17	67
9	3	4
10	3	6
11	1	1
12	1	1
14	1	1

5.4.1 最遠隣法

100 のクラスターに分類された単語の詳細を確認したところ、37 次元で分類した時に確認した、合計で 499 単語を含む 5 つのクラスターは分割されていた。また、次元数を増やしたことで、クラスター synset は全体的にルート synset から遠くなっている。1 単語 1 クラスターが新しく 14 増えた。新しく 1 単語 1 クラスターとなった単語を以下に示す。

- 薄膜、解、周期、前日、スペクトル、能動、マクスウェル、小川、勢、種々、国際、正当、発揮、熊、自動

この中で、熊、小川は教員の名前から来た単語であることを確認した。

200 のクラスターに分類された単語の詳細を確認したところ、1 単語 1 クラスターが 48 できていた。10 単語以上分類されたクラスターは 20 のクラスターで、全体的に少数の単語が分類されたクラスターが多い。一番多くの単語が分類されたクラスターは 47 単語でクラスター synset は number となっており数字が分類されていた。

5.4.2 ウォード法

100 のクラスターに分類された単語の詳細を確認したところ、最遠隣法よりもクラスター synset の段数はバランスよくなっている。ルート synset から 1 段目のクラスターは単語数は減ったものの 3 単語残った。また、1 単語しか分類されていないクラスターが 1 つ存在し、「国際」という単語だった。

200 のクラスターに分類された単語の詳細を確認したところ、1 単語 1 クラスターが 25 に増えていた。新しく 1 単語 1 クラスターとなった単語を以下に示す。

- 解、磁化、動画、周期、アドバイス、前日、スペクトル、能動、広範囲、生、マクスウェル、小川、勢、三角、付け、国際、正当、コンピュータグラフィック、テストケース、発揮、定常、熊、自動

これらの単語のうち「動画」以外は、最遠隣法でも 200 のクラスターに分類した時に 1 単語 1 クラスターとなっていた単語だったのを確認した。

5.4.3 次元数を増やした場合の結果と考察

次元数を増やすことで、考えた通りクラスター synset はルート synset から遠くなった。また、1 単語 1 クラスターとなる単語を確認した。1 単語しか分類されていないクラスターは両方とも増加した。次元数を増やすことでクラスター synset が 1 単語 1 クラスターも増加することがわかった。1 単語 1 クラスターになった単語は、両方の方法で共通する単語が多かった。

二つの方法を比較した結果、ウォード法の方がルート synset に近いクラスター synset が多いが、1 単語 1 クラスターは少なく、よく単語を分類できていると判断した。本手法ではウォード法を採用することにした。

第6章　実験2:クラスター数増加させた場合の類似度の変化

5節の結果から、圧縮する次元数を増やすことで、クラスターsynsetは抽象的な概念から具体的な意味の概念になり、そのクラスターsynsetに分類された単語同士は近い概念に所属することになることがわかった。本手法で次元数を増やすことで文書間の類似度がよりよく求められると考え、次元数を増やした場合の文書間の類似度の変化を調べた。

文書間の類似度の結果を本手法のみで判断することは難しいため、比較対象としてLSIを利用した類似度計算の結果を使用するが、3.4.1節で述べたように日本語WordNetを使用する本手法では、文書に含まれている名詞と日本語WordNet登録されている名詞に差があることがわかっている。そこで、文書に含まれる全ての名詞を利用した場合、使用する名詞を日本語WordNet登録されている名詞のみの場合の2種類のLSIを利用した類似度計算の結果を比較対象として使用することにした。5節で使用したファイルと4.2.1節で分類したテキストをファイルに分けたものを対象に類似度計算を行う。圧縮する次元数としては、累積寄与率が80%になる次元数の他に、シラバスのファイルと4.2.2節の学科と高専のテキストでは含まれる名詞の数がかなり違うため、次元数をシラバスでは50単位、学科のテキストでは500単位で次元数を変えていき、類似度の変化を確認していくことにした。

6.1　岐阜高専_電気情報工学科の類似度

5節で使用していた文書を対象に次元数を増やし類似度計算を行った。このファイルはあまり単語数がないため、50ずつ次元数を増やし結果を確認した。比較対象として、以下のFigure 6.1に文書に含まれる全ての名詞でLSIを利用した類似度計算を行った場合の結果をFigure 6.2に辞書に登録されていた名詞でLSIを利用した類似度計算を行った場合の結果を示す。辞書登録がある名詞での累積寄与率が80%を超えるのは36次元だった。以下のFigure 6.3, Figure 6.4, Figure 6.5, Figure 6.6にそれぞれ37, 100, 200, 300次元で圧縮した本手法による類似度計算の結果を示す。

6.1.1　類似度計算の結果

LSIを利用した類似度計算の結果

Figure 6.1, Figure 6.2を確認した結果、それぞれ違う結果となった。また、全体的には同じ名前の科目で分類されていたりする以外は比較的高い地点で分類されている。高い地

点で分類されると類似度は低くなるので、いくつかの類似度の高い教科とそれ以外では類似度が低いという結果になった。

次元数 37 の結果

Figure 6.3 を確認した結果、LSI を利用した類似度計算の結果より全体的に高さが低くなっていることがわかった。教科の分類については、電気系の科目のみや実験と研究が分類されるなど類似性のありそうな科目で分類されているが、LSI を利用した結果では分類されていた応用数学は近くに分類できなかった。

次元数 100 の結果

Figure 6.4 を確認した結果、次元数 37 の結果に比べ樹形図の高さが全体的に高くなった。次元数 37 では高さ 0.5 で 19 のクラスターに分類されているが、次元数 100 では 27 のクラスターに分類された。高さが高い地点で分割されるほど類似性が低くなるので、クラスター間の類似度は低くなっていると判断できる。教科の分類としては、最初に実験と研究、電気材料、プログラムの科目と他の科目で分かれたが、電気系と情報系の科目が混じるクラスターが多くあまり分類できているとは思えなかった。

次元数 200 の結果

Figure 6.5 を確認した結果、高さ 0.5 で 39 のクラスターに分類されるようになった。教科の分類として電気系の科目群が片側に集まった。また、これまでバラバラだった応用数学が近くに集まるなど次元数 100 の時よりもよく類似性を求められたように思う。

次元数 300 の結果

Figure 6.6 を確認した結果、高さ 0.5 で 42 のクラスターに分類されるようになった。教科の分類としては、次元数 100 と同じように実験と研究、プログラムの科目と他の科目で分かれた。他には特徴的な分類はなく、次元数 200 の結果の方がよく類似度を求められたように思った。また、250 次元より大きい次元数では全ての結果で、実験や卒業研究のクラスターとそれ以外で分類されるようになっていた。

結果の考察

次元数を増加させることで類似度が変化することが確認できた。また、次元数を増加させることによって文書間の類似性が下がっている傾向にあることが分かった。教科間の関係性については、次元数 200 の時の結果が一番よく類似度を求められた。次元数を増やすことによる文書間の類似度の変化について明確にすることはできなかった。他の文書では結果が変わる可能性があると考え他の文書でも同様の実験を行うことにした。

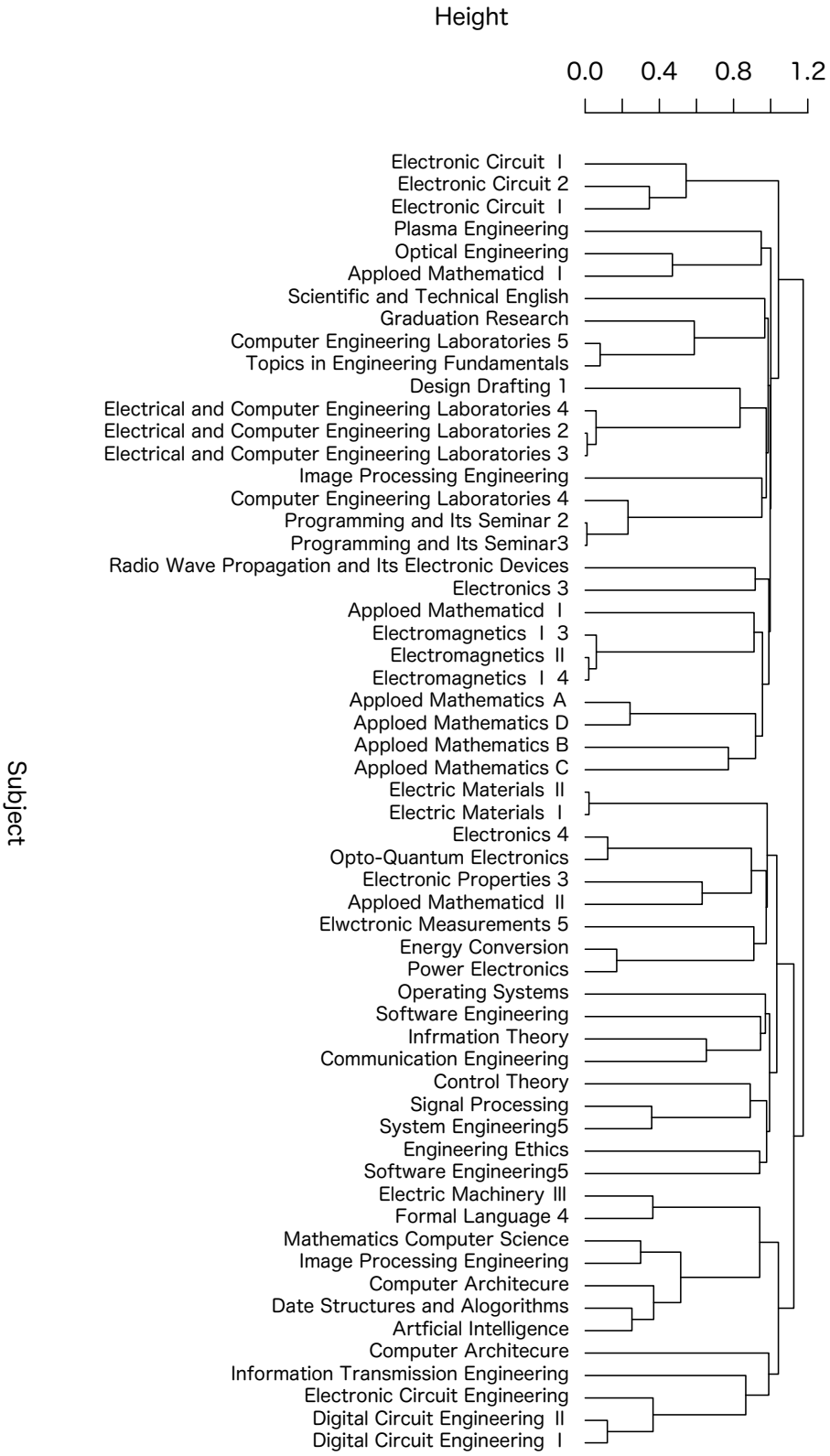


Figure 6.1: Result of Similarity calculation by cos similarity using LSI for dimensional compression

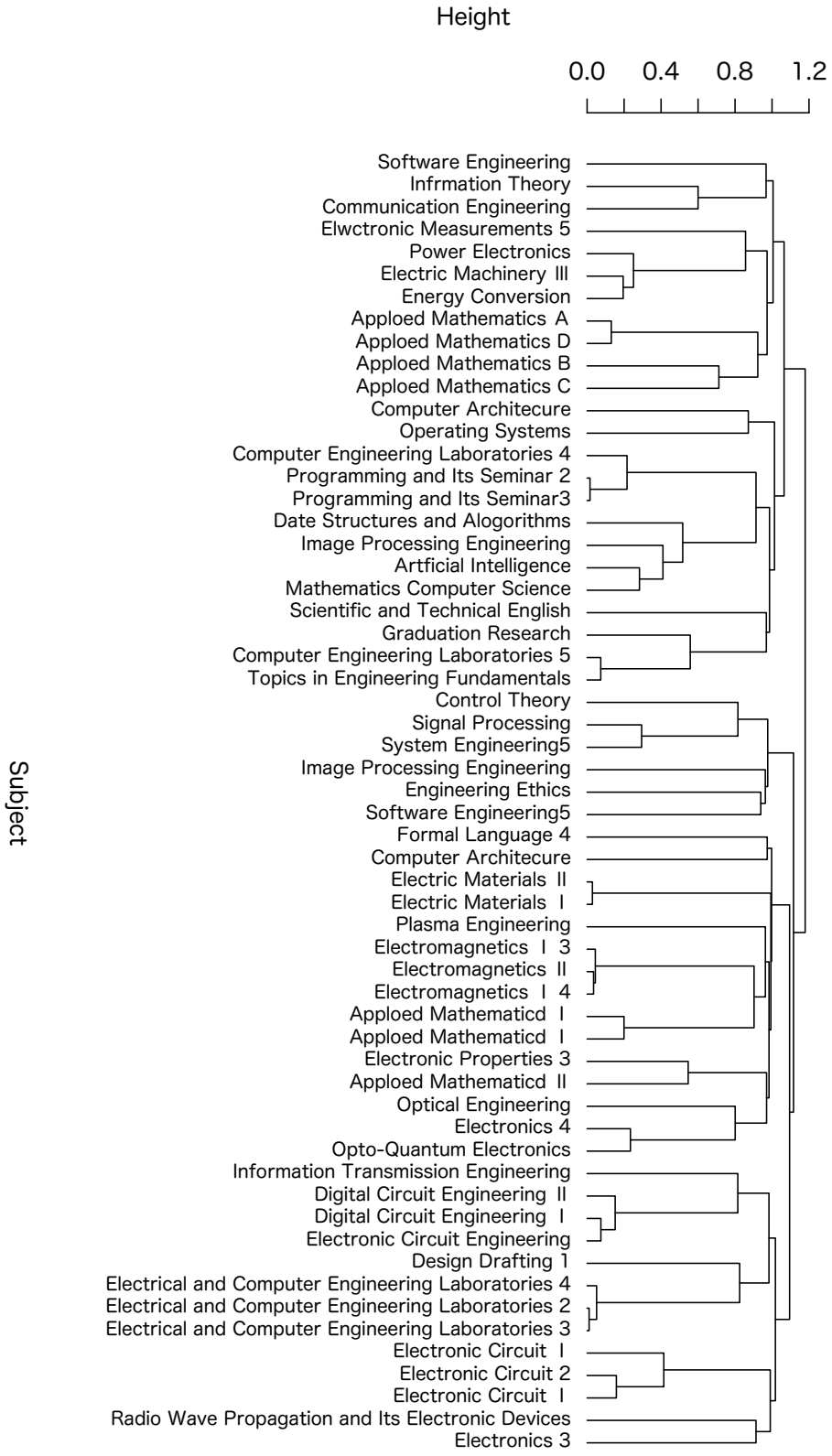


Figure 6.2: Result of similarity calculation by cos similarity using small word with dimension compression LSI

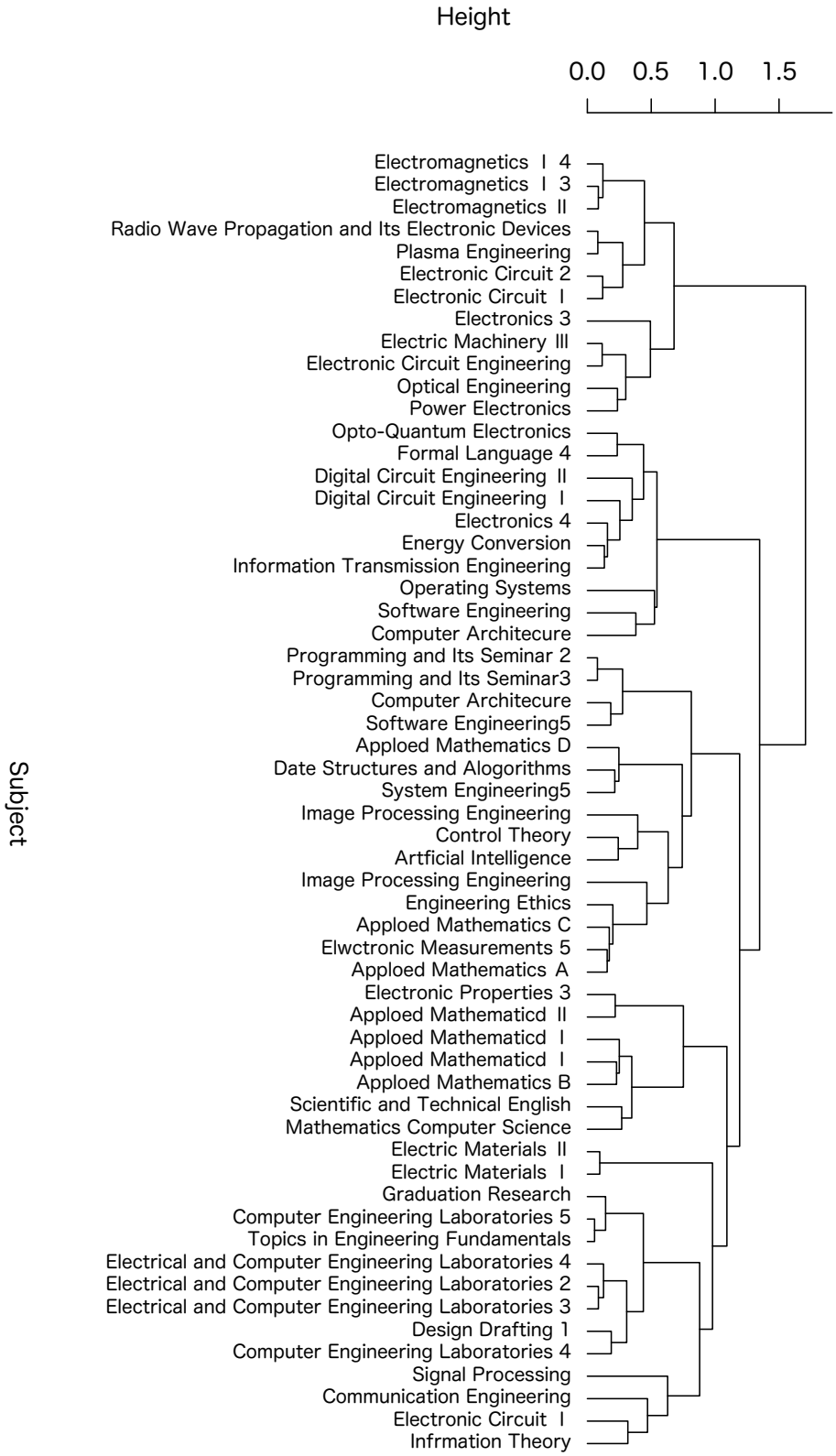


Figure 6.3: Result of using dimensionality reduction by cluster analysis using conceptual distance in 37 dimensions

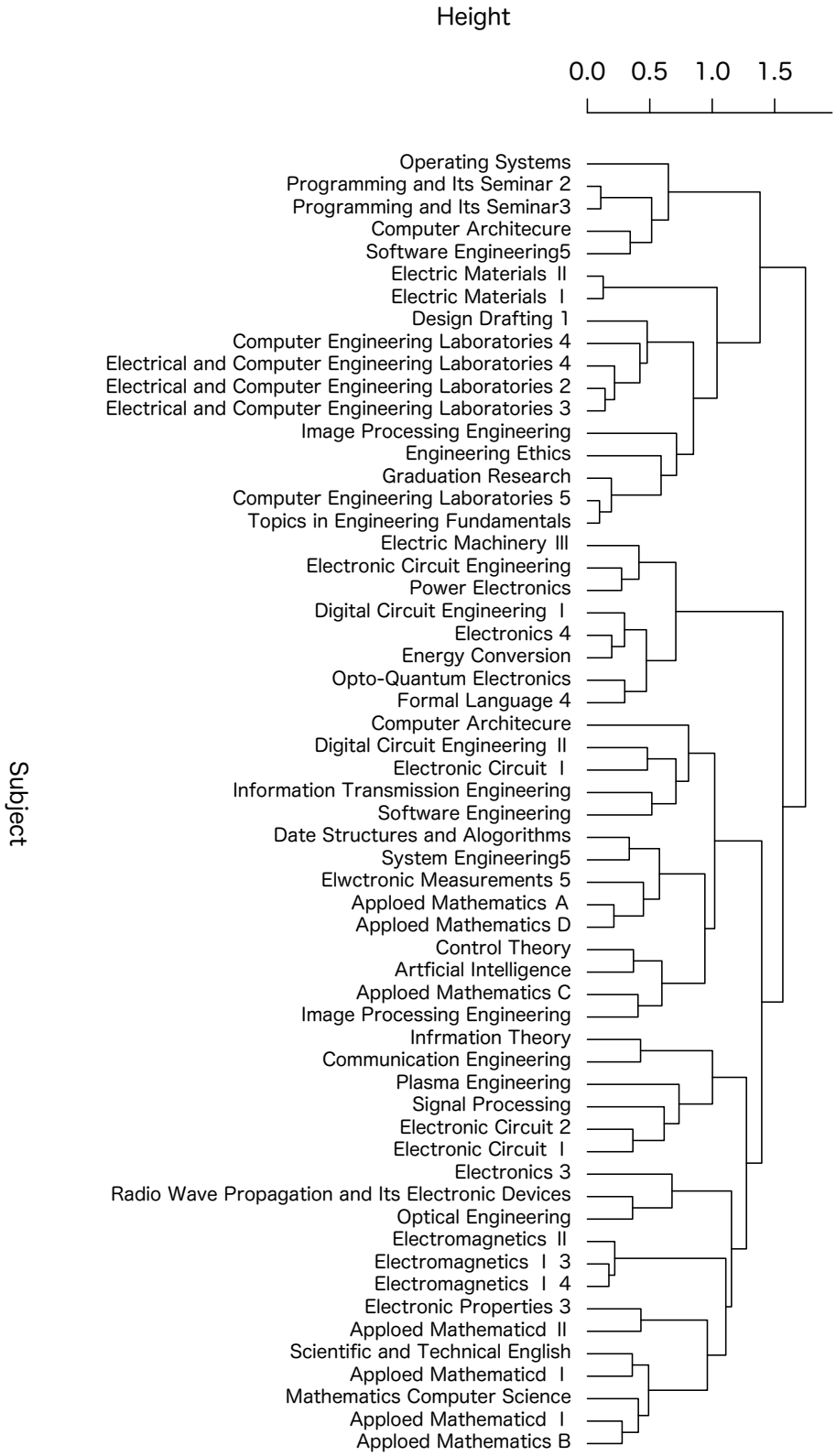


Figure 6.4: Result of using dimensionality reduction by cluster analysis using conceptual distance in 100 dimensions

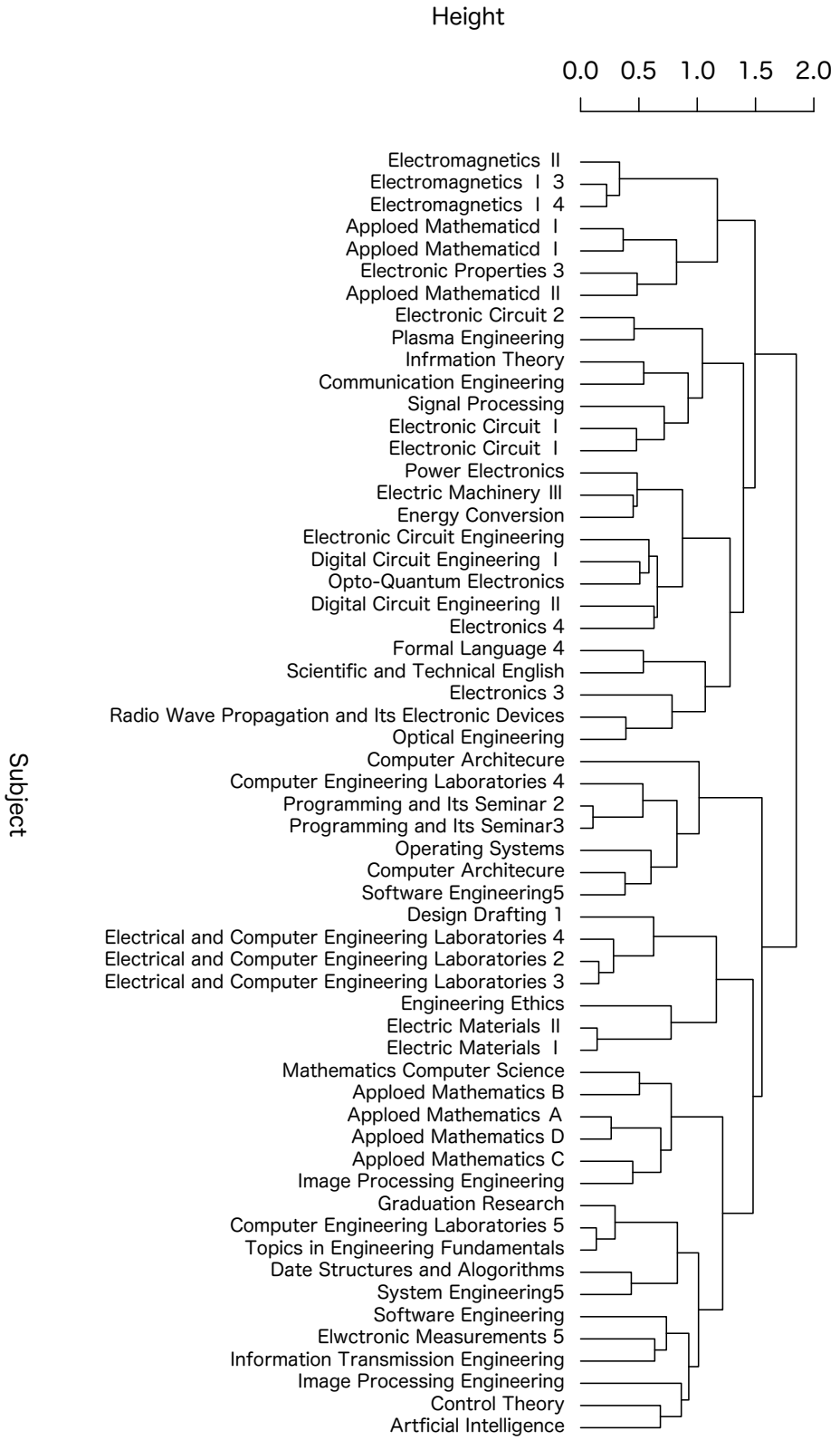


Figure 6.5: Result of using dimensionality reduction by cluster analysis using conceptual distance in 200 dimensions

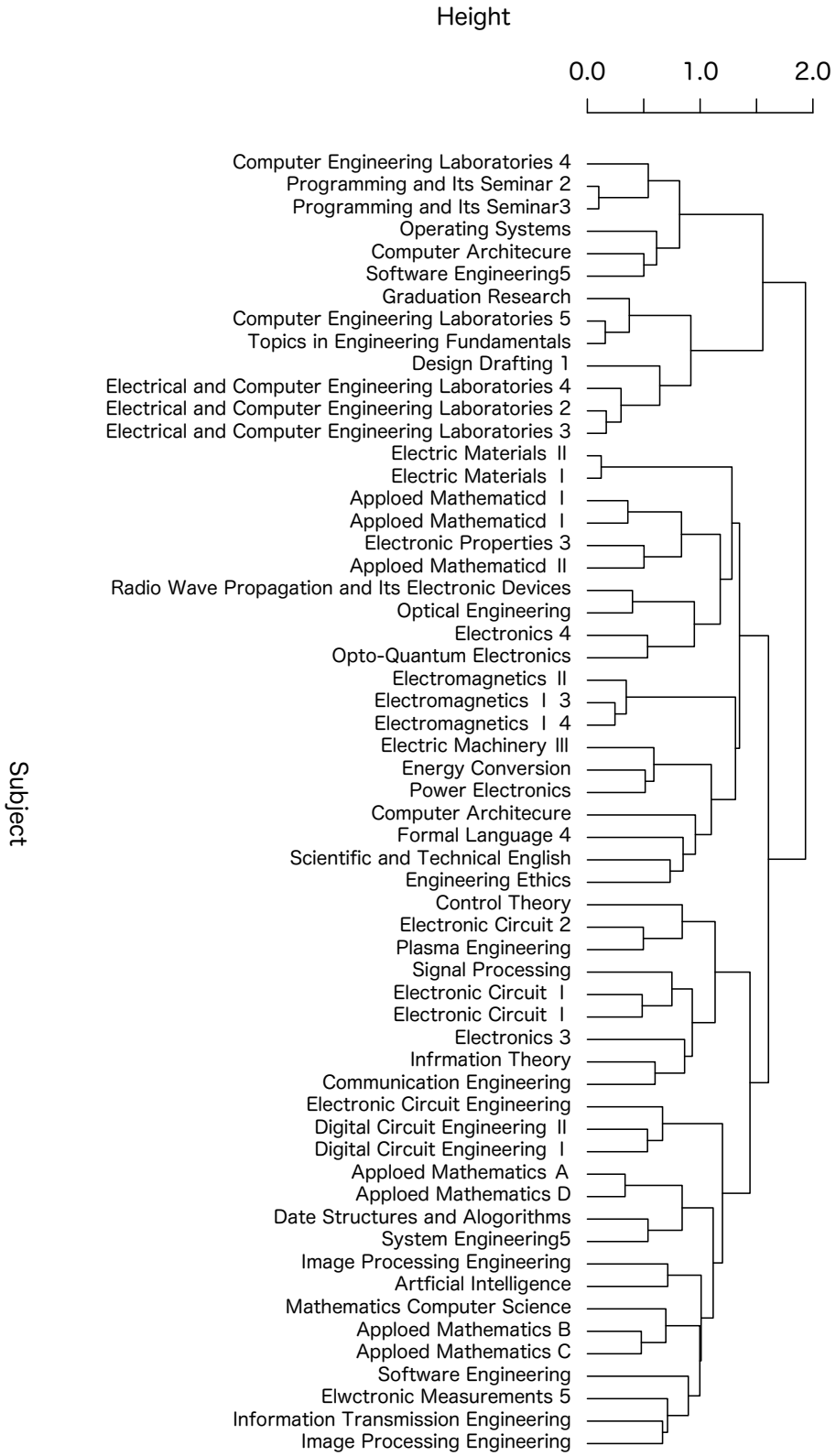


Figure 6.6: Result of using dimensionality reduction by cluster analysis using conceptual distance in 300 dimensions

6.2　シラバス間の類似度

教科のシラバスをいくつかルールを決めフォルダに分類した。分類したフォルダの文書を対象に、次元数を変えながら類似度計算を行った。以下に類似度計算に使用するために製作した各フォルダのルールについて示す。

1. 5 高専から国語、数学、世界史、英語の 4 科目を集める。
2. 3 高専から、岐阜高専の電気情報科の科目である電気磁気学、オペレーティングシステム、画像処理、信号処理、数学 A に近い科目を集める。
3. 5 高専から、 化学、生物、物理の 3 科目を集める。

以下に 6.2 節の一番目の文書を対象に類似度計算を行った結果を示す。比較対象として、以下の Figure 6.7 に文書に含まれる全ての名詞で LSI を利用した類似度計算を行った場合の結果を Figure 6.8 に辞書に登録されていた名詞で LSI を利用した類似度計算を行った場合の結果を示す。以下の Figure 6.9, Figure 6.10, Figure 6.11 に次元数 13 と 50 と 100 でそれぞれ次元圧縮した場合の結果の樹形図を示す。このフォルダには 1641 の名詞が含まれており、このうち 985 の名詞が辞書に登録されていた。また、累積寄与率は、文書に含まれる全ての名詞と辞書に登録されていた名詞の両方の場合で、13 次元で 80%を超えた。

6.2.1　類似度計算の結果

LSI を利用した類似度計算の結果

Figure 6.7 を確認した結果、数学、国語、歴史、英語の 4 科目の群と岐阜高専の 3 科目の 5 つに大きく分けられた。また、舞鶴の国語国文は国語科目として分類された。岐阜高専の科目が集まった理由としては、シラバスを確認したところ「AL のレベル」という言葉が共通して繰り返し書かれていることがわかった。岐阜高専の世界史には「AL のレベル」は書かれていなかった。TfIdf は出現数によって値が大きくなるので、繰り返し出現したこの言葉が岐阜高専の科目を集める結果になったと考えられる。Figure 6.8 の結果では、岐阜高専の数学 A と鹿児島の応用数学が近くなった。辞書に AL という単語は登録されていないため、岐阜高専の科目どうしの類似度が少なくなったと考えられる。

次元数 13 の結果

Figure 6.9 を確認した結果、大きく 3 つに分けてみると世界史科目と数学科目、国語科目と英語科目、歴史科目と数学科目と英語科目で分類されている。各科目が混じって分類されておりよく類似度を求められていない。LSI の方が科目ごとに分類できている。

次元数 50 の結果

Figure 6.10 を確認した結果、世界史科目とそれ以外に分かれた。また、数学科目 3 つでできたクラスターができたものの、残りは英語、国語、数学科目が混じる分類となった。次元数を増加させたことで、世界史科目が分類できるようになった。

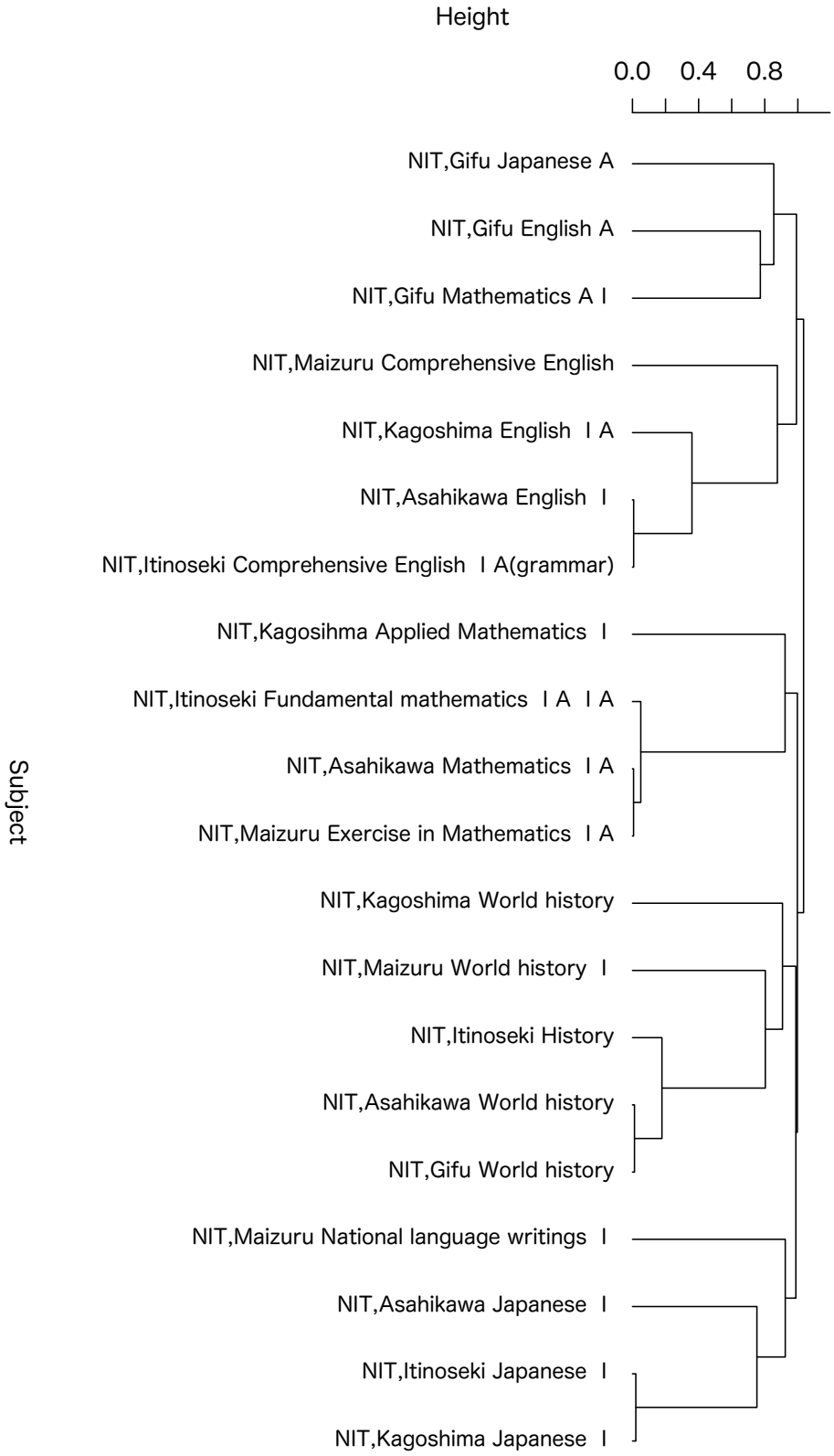


Figure 6.7: Result of Similarity calculation by cos similarity using LSI for dimensional compression

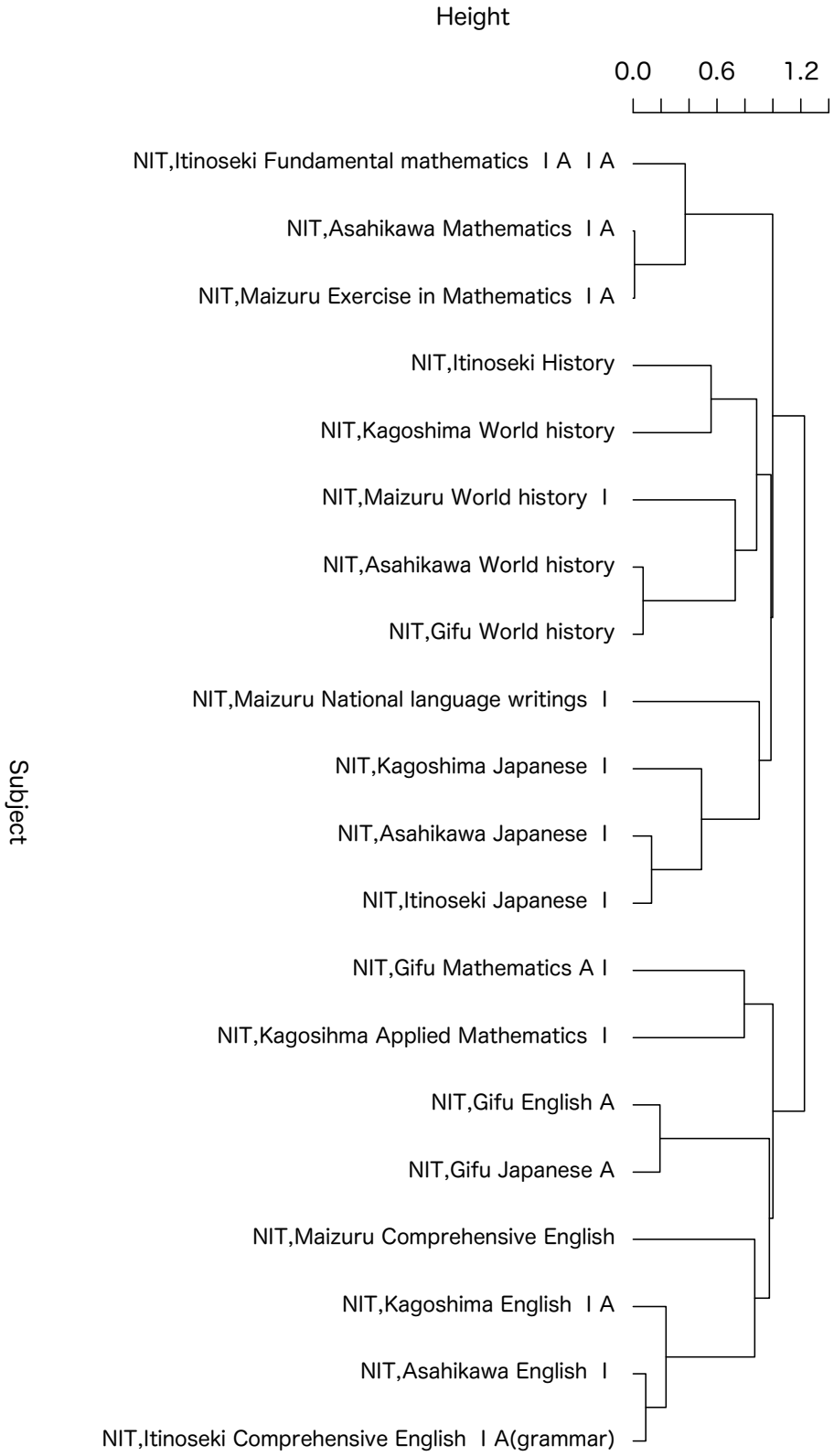


Figure 6.8: Result of similarity calculation by cos similarity using small word with dimension compression LSI

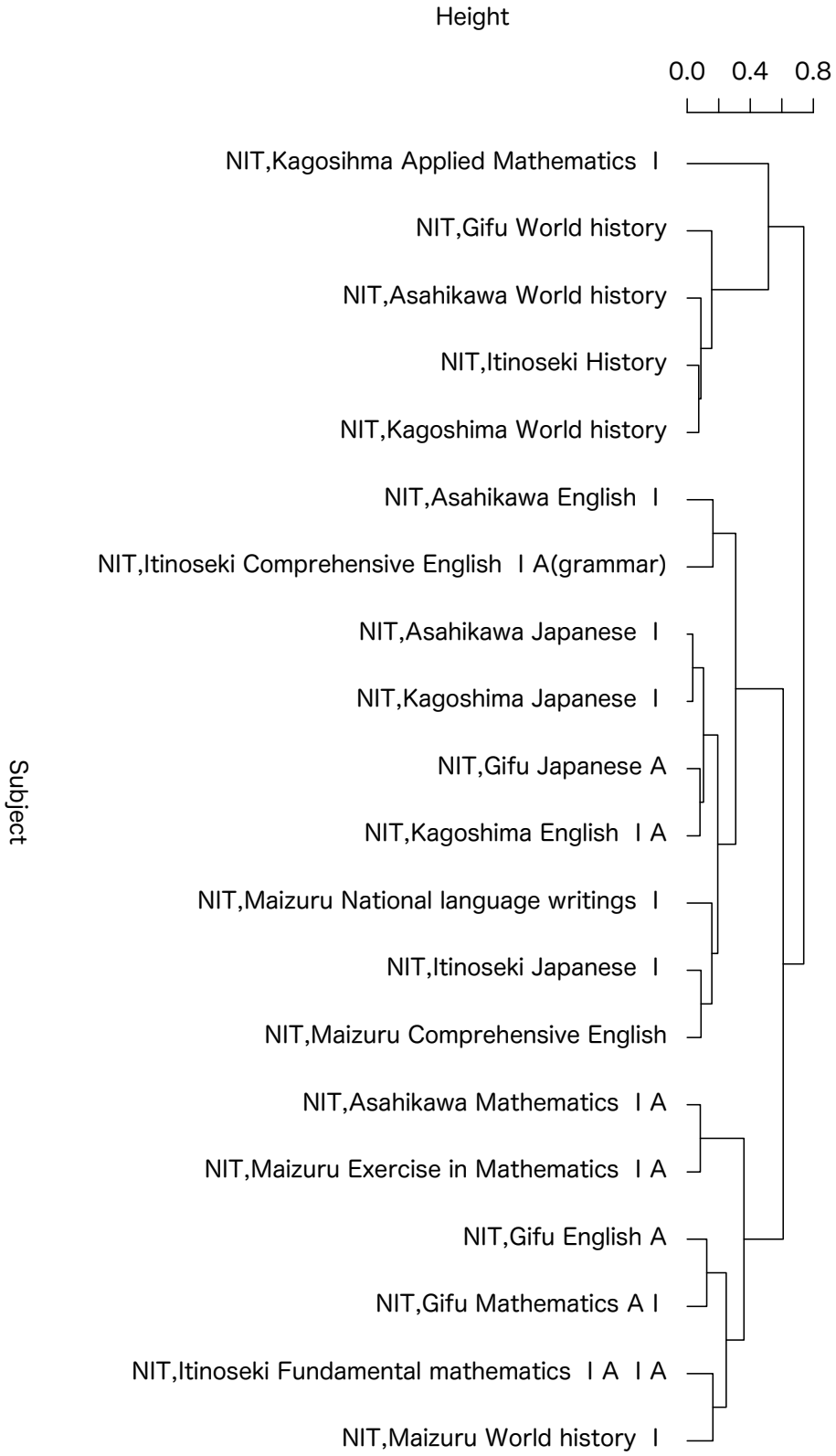


Figure 6.9: Result of using dimensionality reduction by cluster analysis using conceptual distance in 13 dimensions

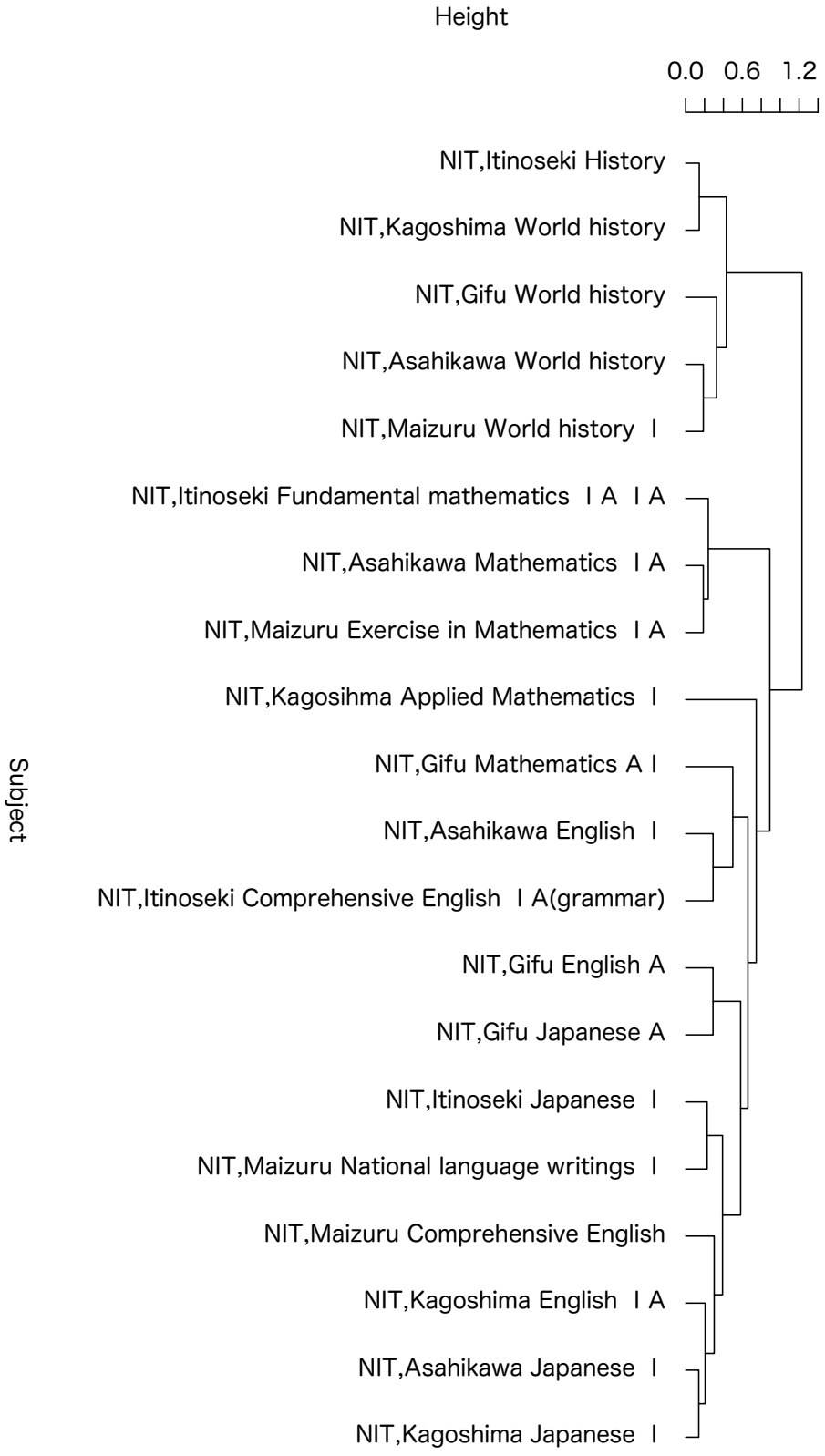


Figure 6.10: Result of using dimensionality reduction by cluster analysis using conceptual distance in 50 dimensions

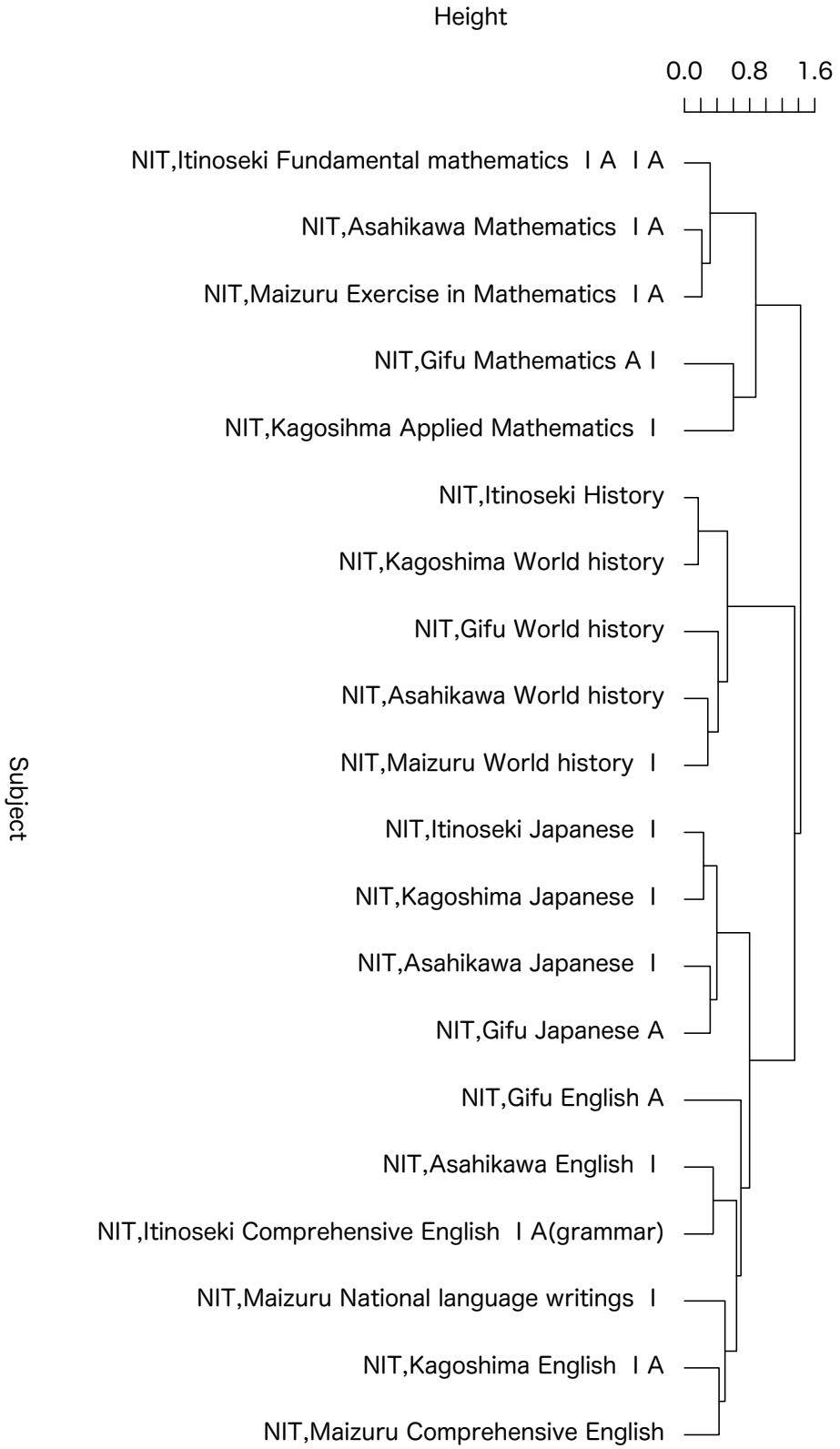


Figure 6.11: Result of using dimensionality reduction by cluster analysis using conceptual distance in 100 dimensions

次元数 100 の結果

Figure 6.10 を確認した結果、舞鶴高専の国語国文を除き、4 つの科目群ごとに分類できるようになった。この結果、岐阜高専の科目を本手法では分類できることが分かった。このファイルでは、次元数を増やすほどよく分類できるようになっていくことが分かった。

次元数 100 以上にした場合の結果

次元数を 950 まで増加させ結果を確認したところ、次元数が 300 から 750 まで樹形図には高さが少しずつ高くなる以外変化がなかった。次元数を 800 以上にすると英語科目と国語科目が混じって分類されるようになった。また本手法では、LSI で国語科目として分類された、舞鶴高専の国語国文が他の国語科目ではなく英語科目と類似性が高かったため、シラバスを確認したところ英語で履歴書や自己紹介文を書く科目ということが判明した。

6.2.2 結果の考察

類似度計算の結果を確認したところ、4 つの科目群に分類することができた。また、LSI を利用した類似度計算の結果とは違う分類をした。LSI を利用した類似度計算と比較すると、本手法の 4 科目群に分類した方がよく分類できていると思われる。次元数を増加させていくと樹形図の変化が少なくなる傾向があることを確認した。舞鶴高専の国語国文が国語の科目ではなく英語科目とわかった。

他のファイルでの結果

本手法では、次元数を増加させていくと樹形図の変化が少なくなる傾向があることが分かったので、6.2 節の他のフォルダでも同様の実験を行ない次元数を増加させた場合の樹形図の変化を確認した。二番目のフォルダでは次元数 250 以降、三番目のフォルダでは次元数 400 以降から樹形図の変化がほぼなくなった。また変化がなくなった樹形図は TfIdf を cos 類似度で計算しただけの結果とあまり変わらなかった。本手法で圧縮する次元数は 300 から 250 以下が適切である可能性がある。

6.3 学科間の類似度

4.2.2 節の学科のテキストをいくつかルールを決めフォルダを分類した。分類したフォルダの文書を対象に、次元数を変えながら類似度計算を行った。以下に類似度計算に使用するために分類した各フォルダのルールについて示す。

1. 岐阜高専の機械、電気情報、電子制御、環境都市、建築と近いと思われる専門学科を 20 集め、そこに岐阜の先端融合開発専攻を加える。

2. 一般科目がまとめられた学科のテキストと物質、科学系の専門学科を集め、そこに経営、機械電子、商船学科を 20 学科になるよう集める。

6.3 節の一番目のフォルダを対象に類似度計算を行った。比較対象として、以下の Figure 6.12 に文書に含まれる全ての名詞で LSI を利用した類似度計算を行った場合の結果を Figure 6.13 に辞書に登録されていた名詞で LSI を利用した類似度計算を行った場合の結果を示す。累積寄与率は、文書に含まれる全ての名詞では 8 次元で 80%を超えた。辞書に登録されていた名詞の場合、10 次元で 80%を超えた。累積寄与率が 80%を超える次元が二つでたので、本手法では次元数 10 で類似度計算を行うことにした。以下の Figure 6.14, Figure 6.15, Figure 6.16 に次元数 500 と 1000 でそれぞれ次元圧縮した場合の結果の樹形図を示す。このフォルダには 14807 の名詞が含まれており、このうち 5939 の名詞が辞書に登録されていた。

6.3.1 類似度計算の結果

LSI を利用した類似度計算の結果

Figure 6.12 を確認した結果、高知、岐阜高専では高専ごとに分類された。呉と石川も建築学科以外は近くに分類されている。学科名ではなく高専ごとに類似度が高くなるような分類となった。Figure 6.13 を確認した結果、大きく分類が変わった。大きく分けて建設と環境、電気情報と電子と機械の学科で二つに分類された。電気情報と電子と機械学科の方は、石川と岐阜高専が高専ごとに集まっている。建設と環境の学科の方は、建設と環境で分かれるような分類となっている。名詞が減ることで高専ごとに分類されていたのが、学科で分類されるようになった。先端融合開発専攻は、岐阜高専の他の学科の近くに分類されている。これは 6.2 節の結果とは違う結果となった。

次元数 10 の結果

Figure 6.14 を確認した結果、大きく 3 つに分類された。しかし、LSI を利用した結果のように高専ごとに集まることはなく学科もバラバラなためあまりよく分類できていないと考えられる。

次元数 500 の結果

Figure 6.15 を確認した結果、大きく分けて、建設、環境、機械、電気情報と電子の 4 つに分類された。先端融合開発専攻は、岐阜の電気情報と電子制御の学科の近くに分類された。Figure 6.14 と比べて明らかに分類できるようになった。次元数を増加させることで、類似度計算の結果は明らかに良くなっていることが分かった。LSI を利用した結果と比較すると、Figure 6.13 の結果と近い分類となったが、本手法の方では機械学科を全て分類できた。

次元数 1000 の結果

Figure 6.16 を確認した結果、次元数 500 と比べ大きく分類が変わることはなかった。しかし、石川高専の機械学科が、次元数 500 では他の機械学科に近くに分類されていたのが、

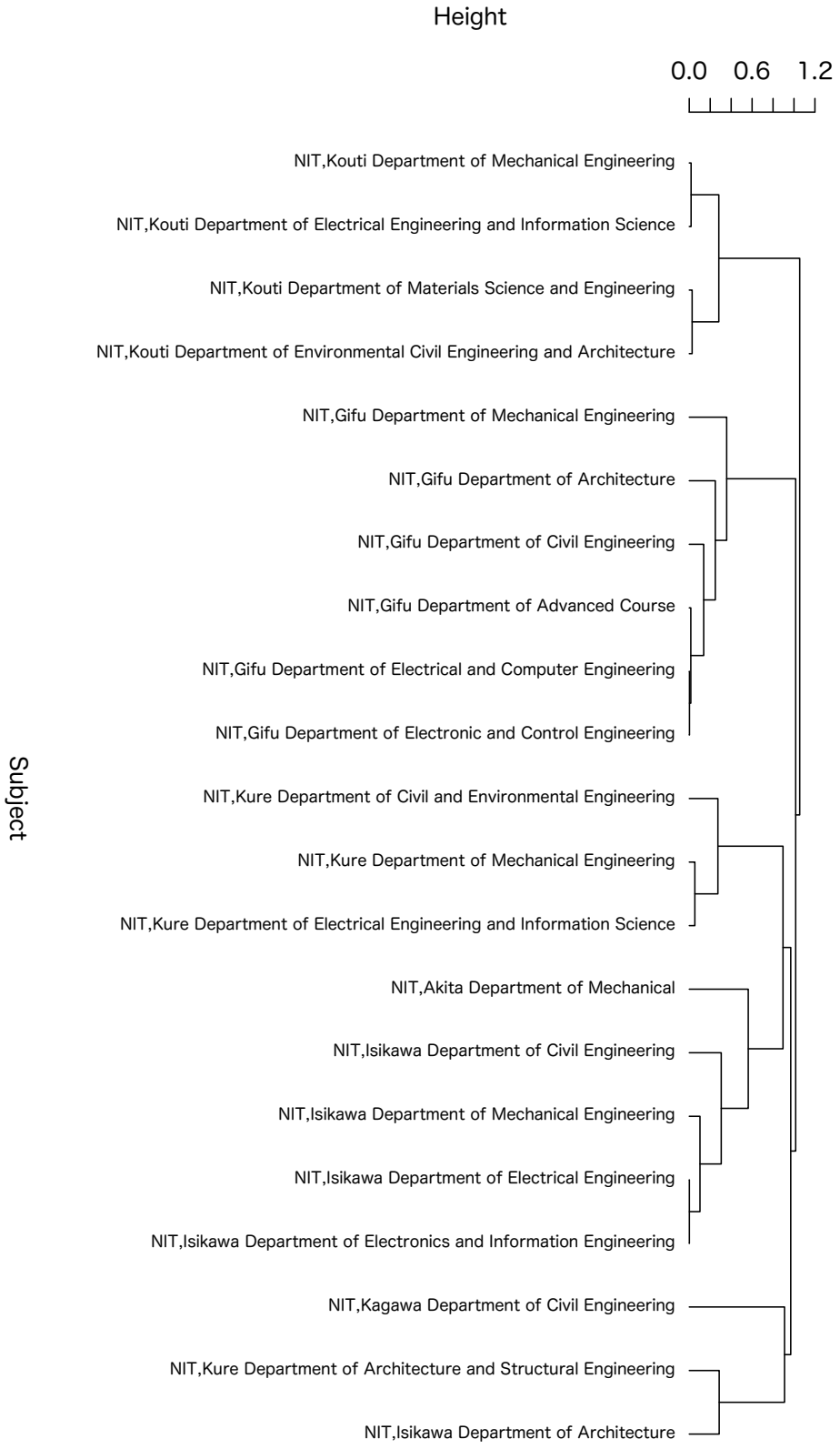


Figure 6.12: Result of Similarity calculation by cos similarity using LSI for dimensional compression

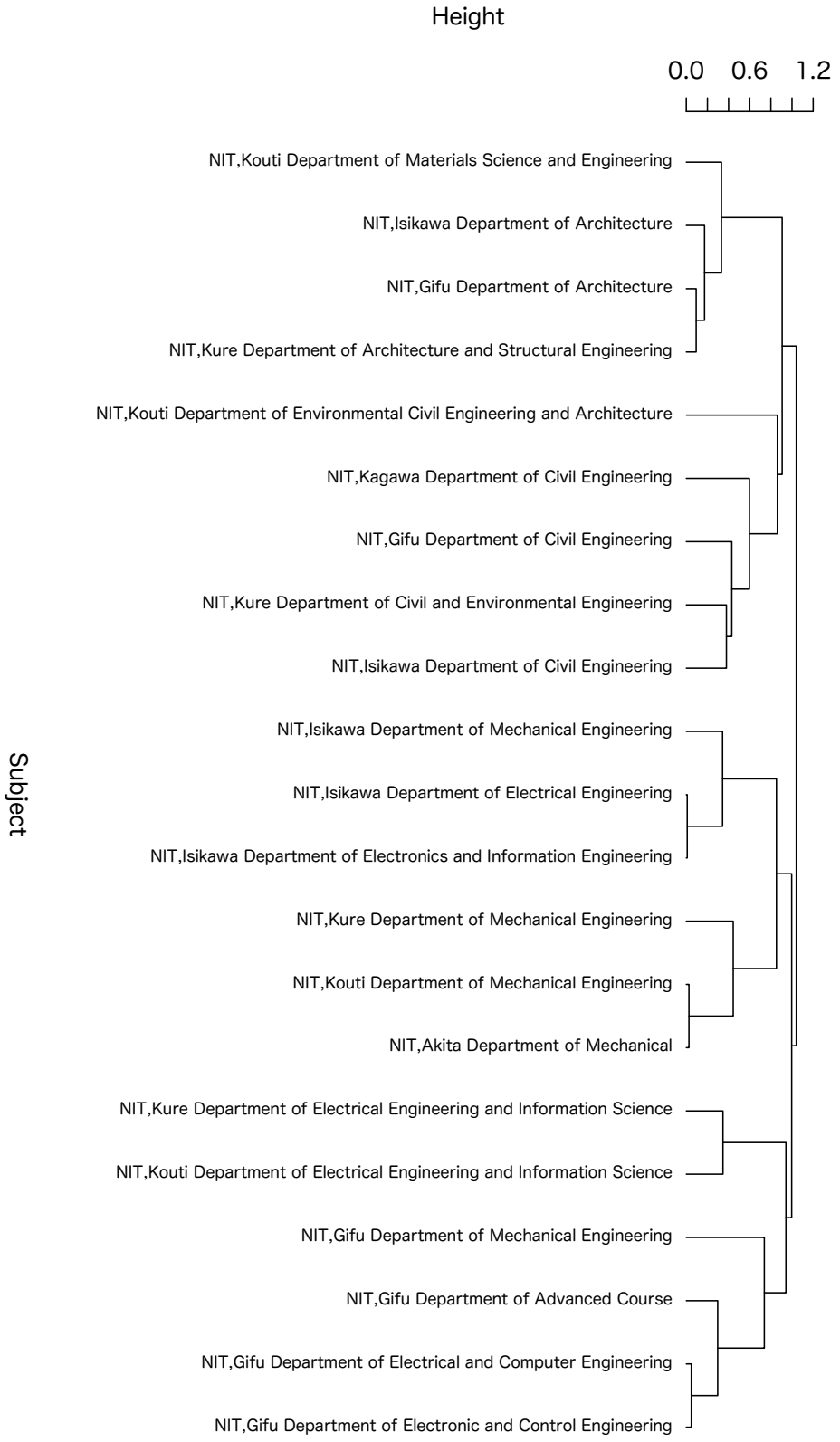


Figure 6.13: Result of similarity calculation by cos similarity using small word with dimension compression LSI

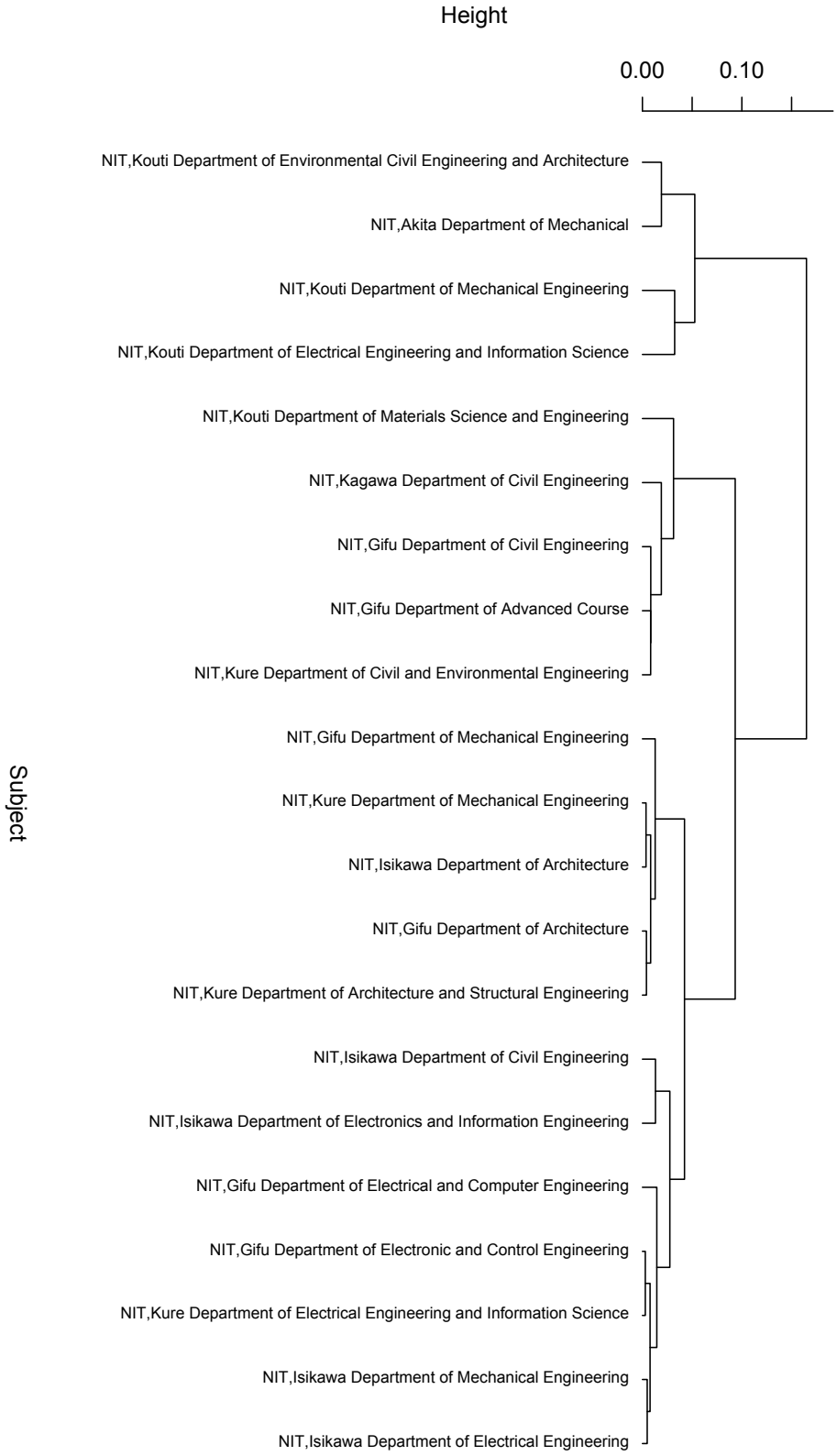


Figure 6.14: Result of using dimensionality reduction by cluster analysis using conceptual distance in 10 dimensions

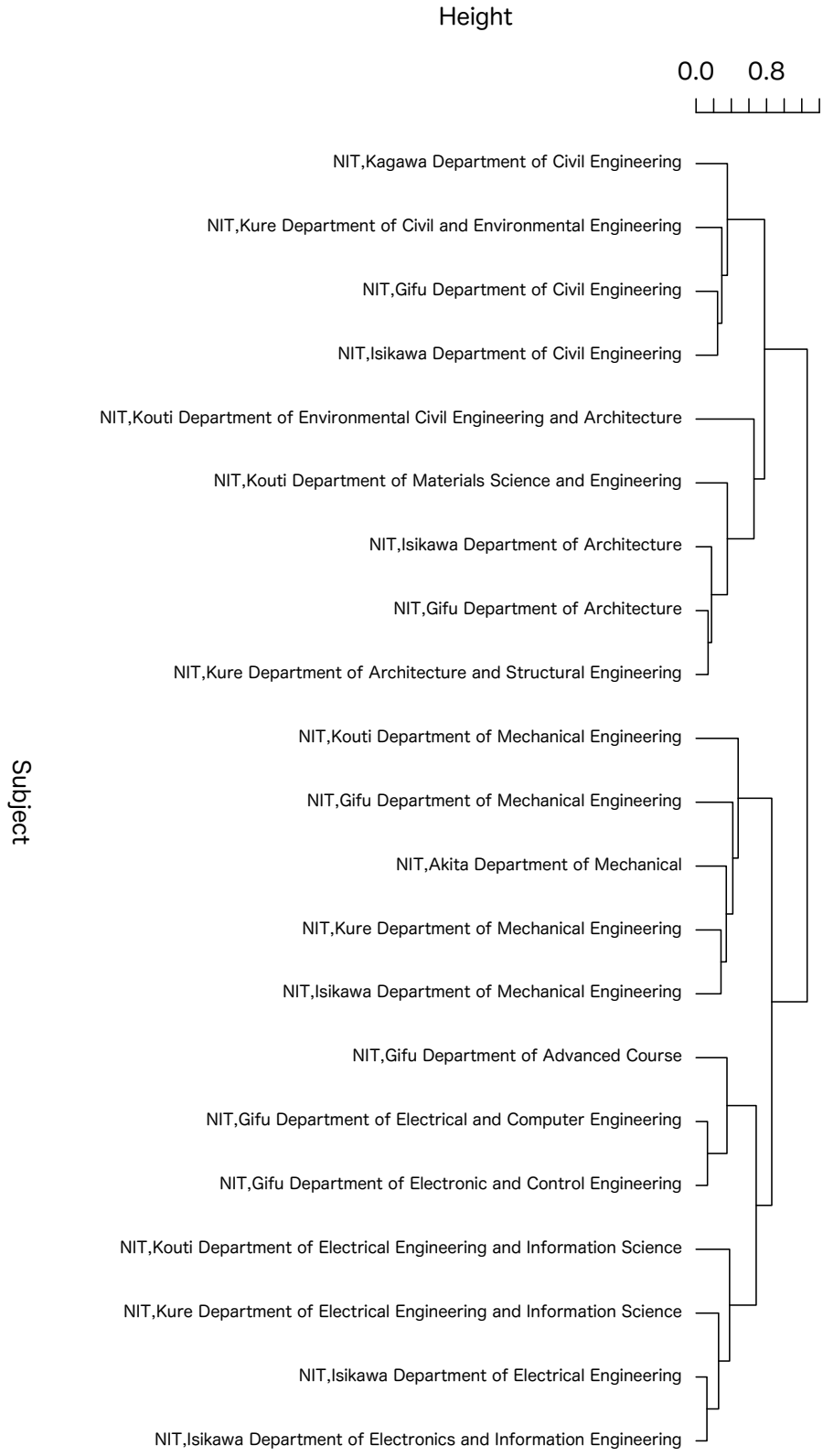


Figure 6.15: Result of using dimensionality reduction by cluster analysis using conceptual distance in 500 dimensions

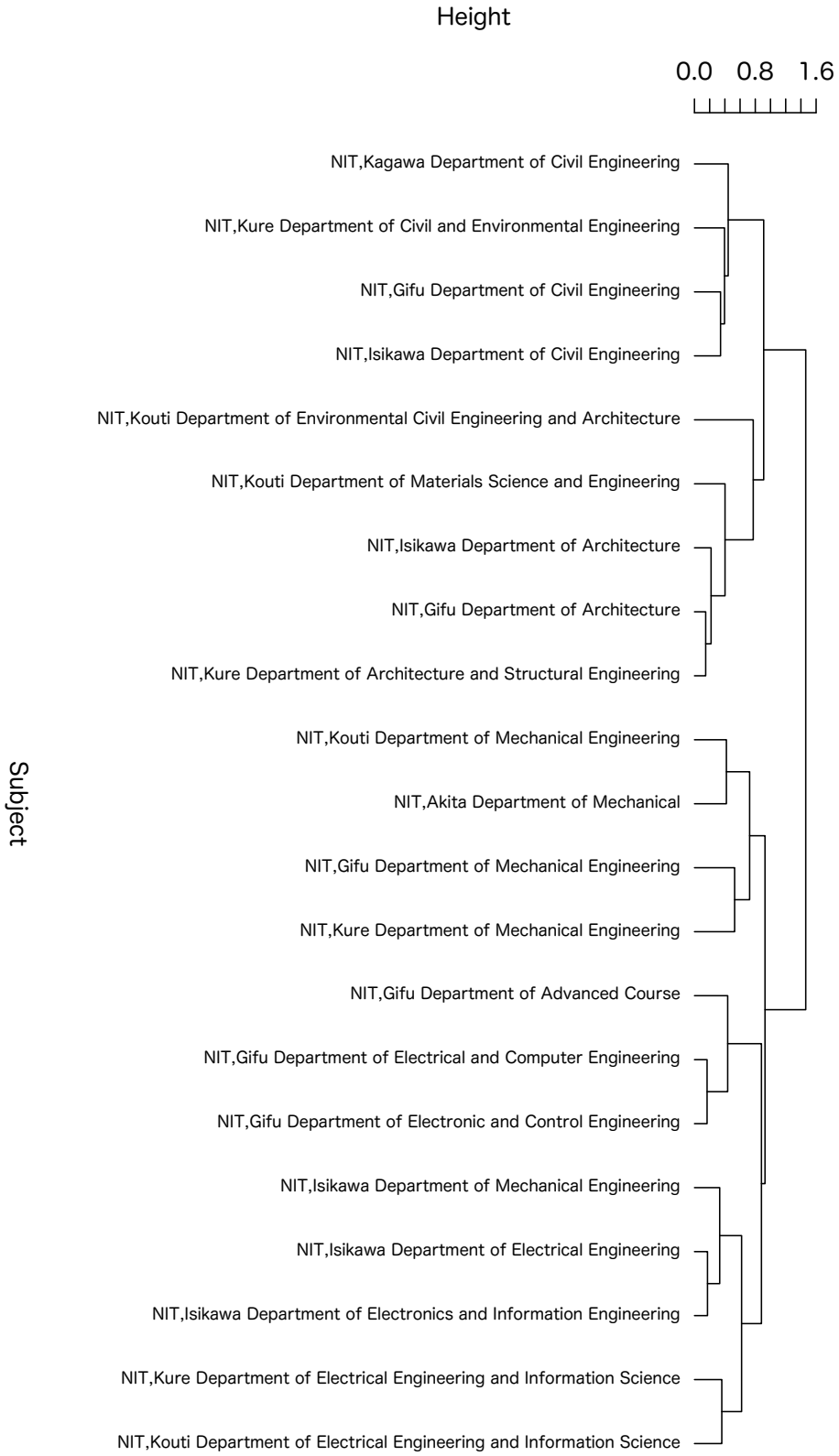


Figure 6.16: Result of using dimensionality reduction by cluster analysis using conceptual distance in 1000 dimensions

石川高専の他の学科と近くなった。先端融合開発専攻は、次元数 500 の結果から変化していなかった。LSI を利用した結果と比較すると、Figure 6.13 の結果に近い分類となった。

次元数 1000 以上にした場合の結果

次元数を 5000 まで増やし確認を行ったが、次元数 1000 の結果から変化はほぼ無いことが分かった。

6.3.2 結果の考察

次元数を増やすことで、明らかに類似度計算の結果が良くなることが分かった。LSI との比較では、辞書登録されている名詞のみの結果に近い分類をすることが分かった。また、本手法では、ある程度まで次元数を増やすと樹形図の変化が少なくなると考えられる。これは、単語をクラスターに分類することによる変化がある段階で少なくなるからと考えられる。

他のファイルでの結果

6.3 節の二番目のフォルダでも同様の実験を行なった。結果として次元数 500 の方が明らかに結果は良くなった。次元数 500 と次元数 1000 では変化がなかった。また、次元数 1000 以上に増やした場合でも大きな変化はなかった。

6.4 高専間の類似度

4.2.2 節の高専のテキストを対象に類似度計算を行った。高専のテキストでは、高専の名称から類似度を判断するのは難しく、51 高専全ての変化を全て比較していくことも難しいため、特徴的な結果のみを比較することにした。比較対象として、以下の Figure 6.17 に文書に含まれる全ての名詞で LSI を利用した類似度計算を行った場合の結果を Figure 6.18 に辞書に登録されていた名詞で LSI を利用した類似度計算を行った場合の結果を示す。累積寄与率は、文書に含まれる全ての名詞では 40 次元で 80%を超えた。辞書に登録されていた名詞の場合、35 次元で 80%を超えた。累積寄与率が 80%を超える次元が二つでたので、本手法では次元数 35 で類似度計算を行うことにした。このフォルダには 68518 の名詞が含まれており、このうち 15250 の名詞が辞書に登録されていた。名詞の数が多いため次元数 500 を求めた後、1000 ずつ次元数を変えて比較する。以下の Figure 6.19, Figure 6.20, Figure 6.21 に次元数 35 と 500 と 1000 でそれぞれ次元圧縮した場合の結果の樹形図を示す。

LSI を利用した類似度計算の結果

Figure 6.17 を確認した結果、3 つの商船、都城と秋田など 7 高専、佐世保と長野、広島と阿南、旭川と小山と群馬の 5 つの分類が確認できた。しかし、他にはあまり類似度が無いという結果になった。Figure 6.18 を確認した結果、大きく分類が変化した。まず、都城

と秋田などを含む 15 高専とそれ以外に分類され、4 つの商船がまとまった。佐世保と長野の近くに分類されている。

35 次元の結果

Figure 6.19 を確認した結果、大きく 4 つに分類された。4 つの商船、北九州と旭川、25 の高専、20 の高専に分類されていた。これまでの結果と違い、次元数が 35 と少ないにも関わらずある程度分類できていると考えられる。LSI を利用した方法の結果と比較すると、全体的に高さが低くなっており高専間の類似度が高くなっていると考えられる。

500 次元の結果

Figure 6.20 を確認した結果、大きく 4 つに分類された。4 つの商船、北九州と旭川のペアに長野と佐世保、14 の高専、29 の高専に分類された。35 次元での結果との大きな違いとして樹形図の高さが高くなっている。北九州と旭川のペアに長野と佐世保が近くなった。14 の高専は、Figure 6.18 の結果に見られた都城と秋田などを含む 15 高専と 12 高専が同じ高専だった。その 12 高専の学科を確認したところ、多くの高専で機械、建設、電気、環境系の学科が存在していた。他の LSI を利用した結果との違いとして大きく 4 つに分類した際の高さが高い、LSI を利用した結果より高専間の類似度が全体的に高くなっている。

1000 次元の結果

Figure 6.21 を確認した結果、大きく 4 つに分類された。4 つの商船、北九州と旭川のペアに長野と佐世保、14 の高専、29 の高専に分類された。大きな分類は変化しなかったが、29 の高専では高専間の類似度が変化した。

次元数 1000 以上にした場合の結果

次元数を増やし結果を確認したところ、大きな分類としては、次元数 500 以降から次元数 10000 まであまり変化はなかった。それ以降は、旭川と佐世保のペアが他の高専と分類された。また、高さの変化はは次元数 35 と 500 の間が一番多くそれ以降は徐々に増加していた。

6.4.1 結果の考察

大きく分けて 4 つに分類することに成功した。LSI を利用した結果と比較したところ、大きく分類する際の高さは本手法の方が高いが、高専間の高さは低くなっていた。次元数を増加させることで、全く変化がなくなるということはなかった。

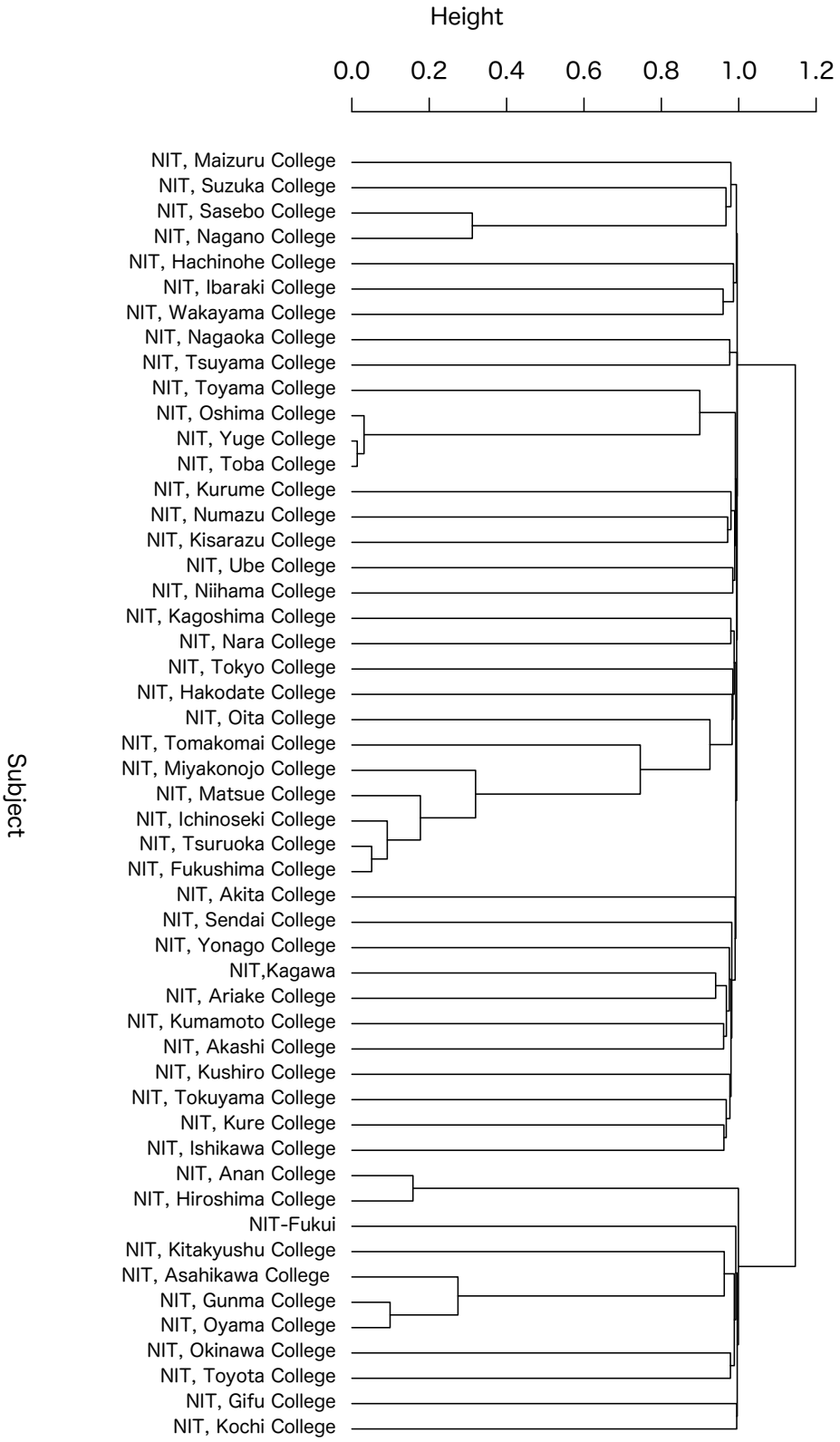


Figure 6.17: Result of Similarity calculation by cos similarity using LSI for dimensional compression

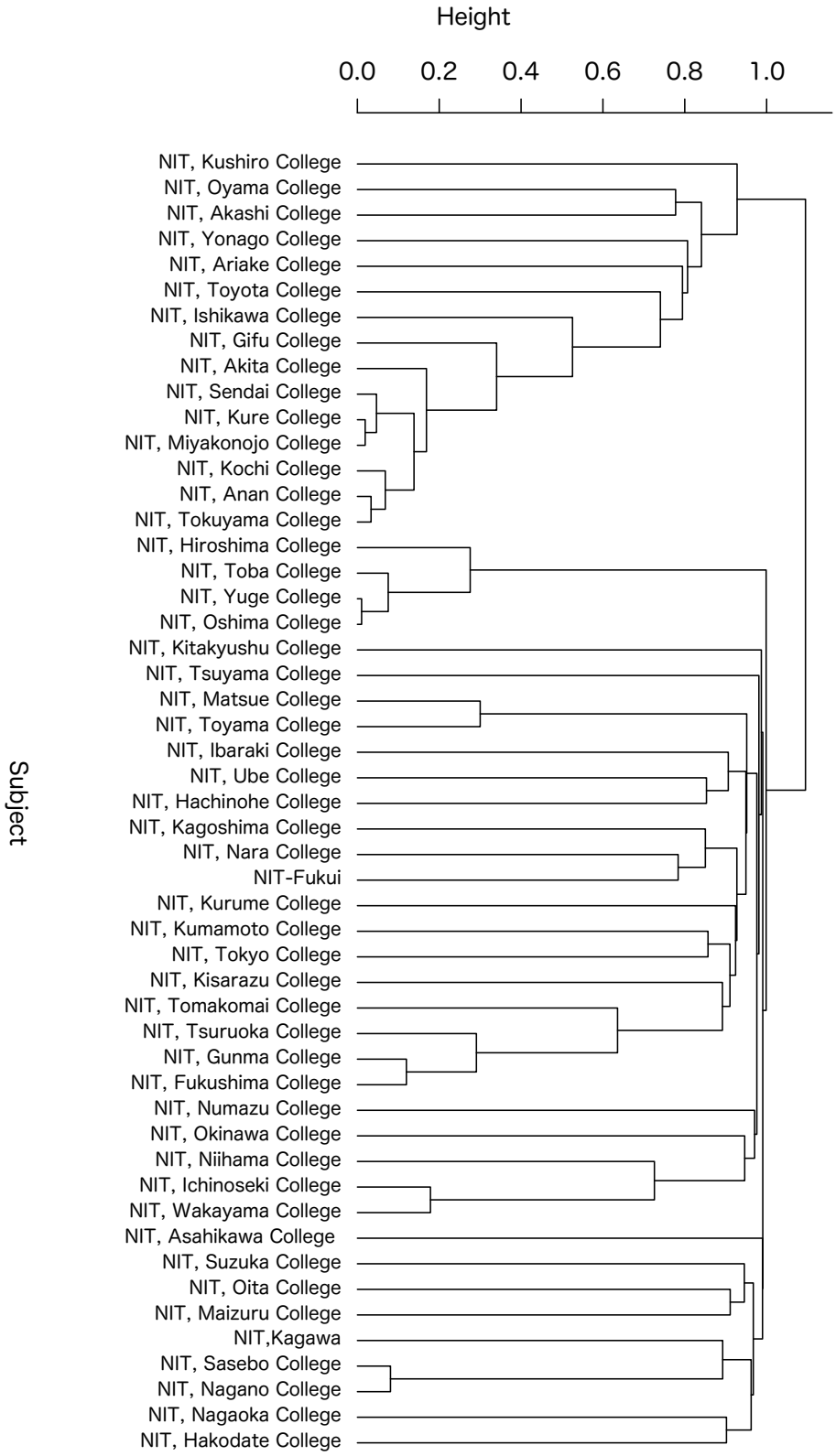


Figure 6.18: Result of similarity calculation by cos similarity using small word with dimension compression LSI

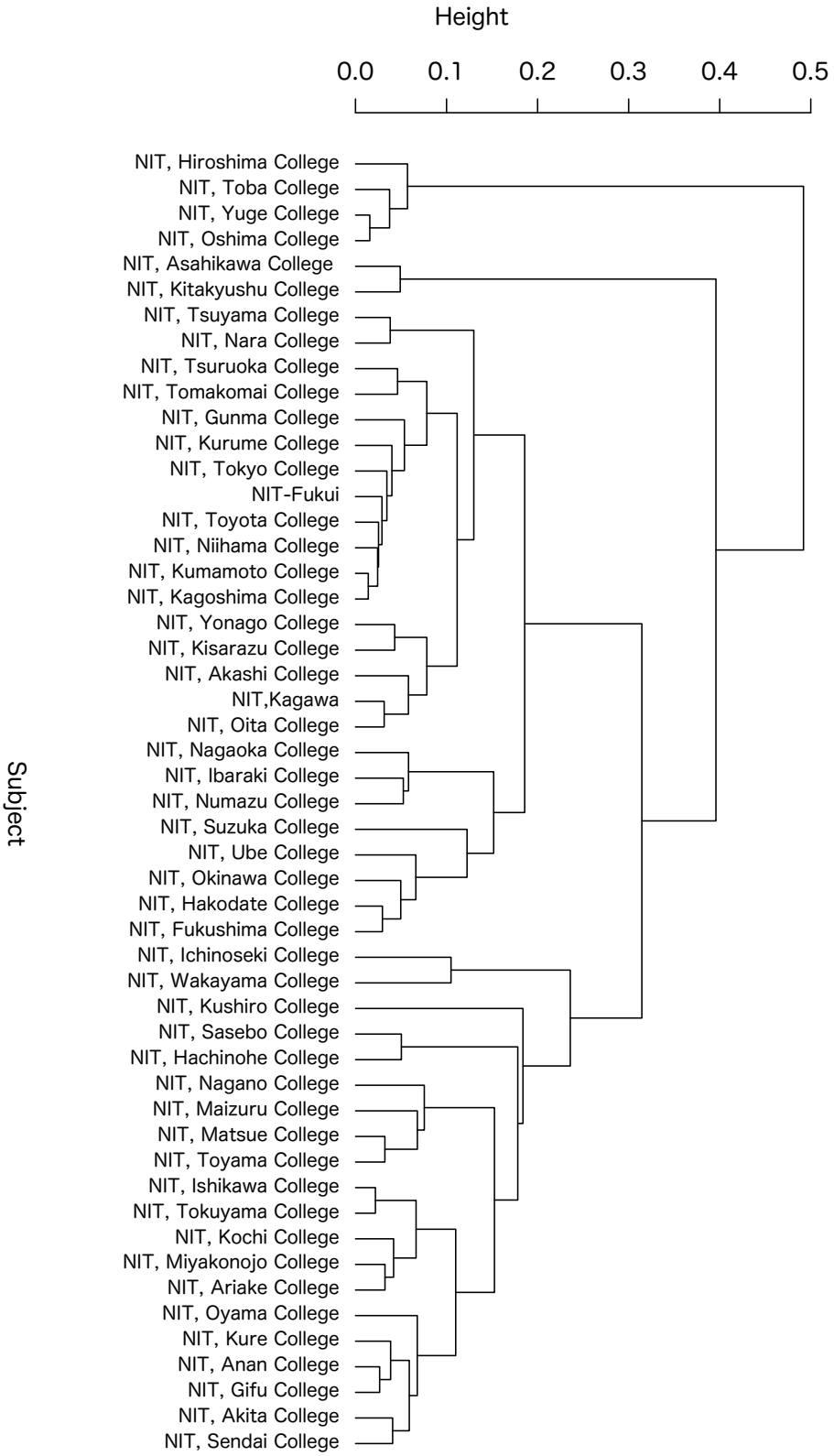


Figure 6.19: Result of using dimensionality reduction by cluster analysis using conceptual distance in 35 dimensions

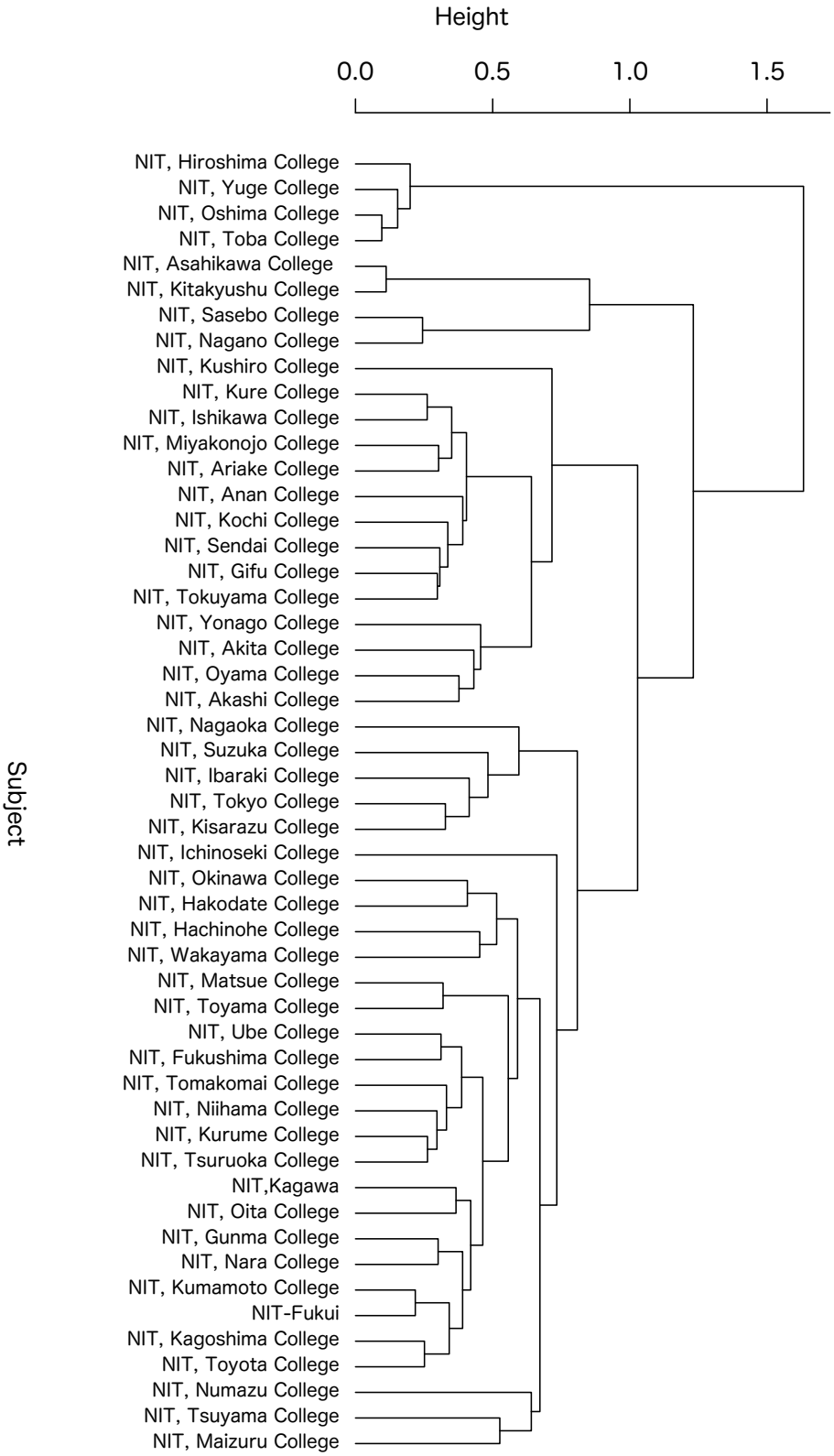


Figure 6.20: Result of using dimensionality reduction by cluster analysis using conceptual distance in 500 dimensions

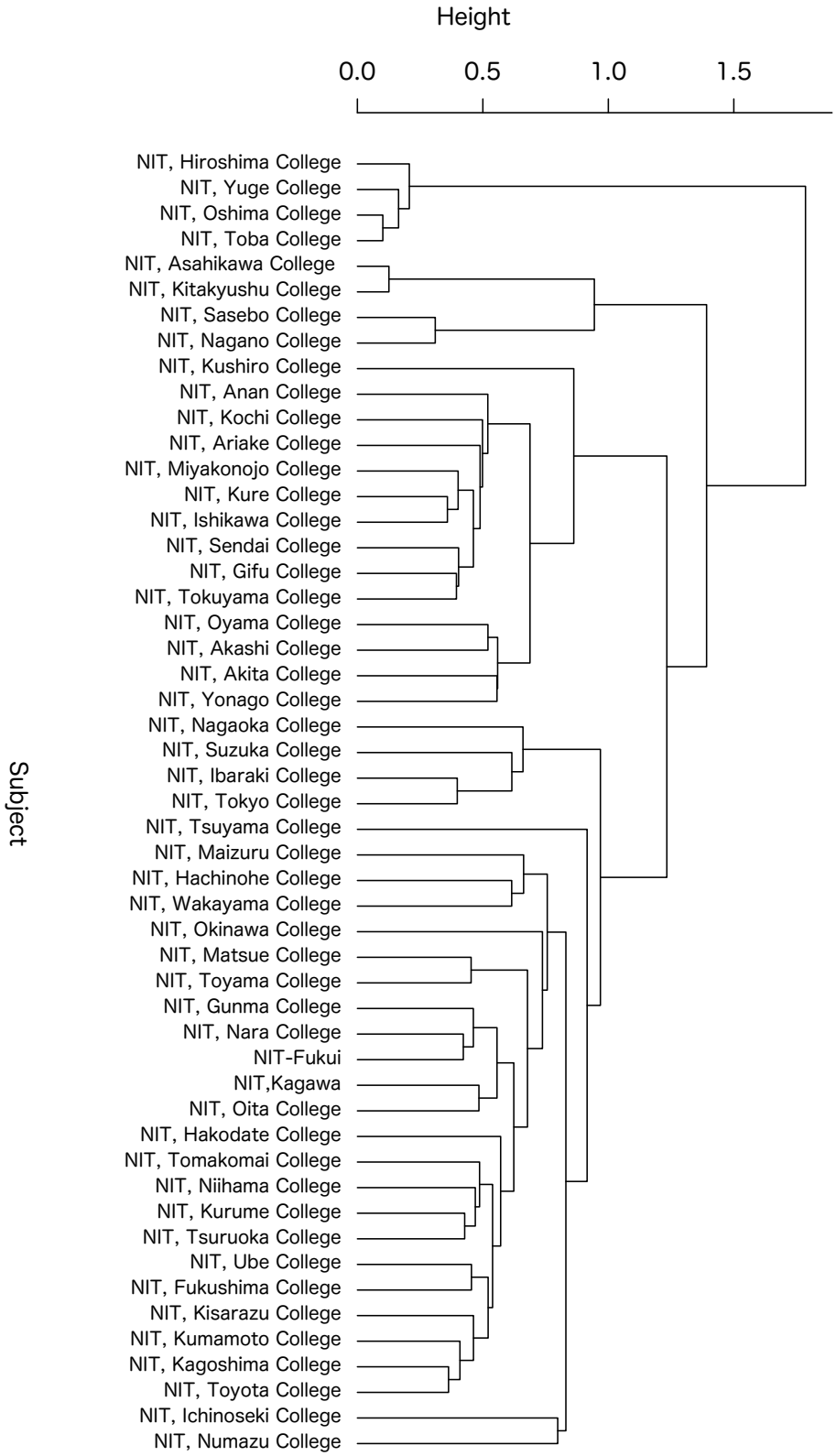


Figure 6.21: Result of using dimensionality reduction by cluster analysis using conceptual distance in 1000 dimensions

6.4.2　本手法の問題

6.4 節の実験を行なった際に発生した問題について述べる。本手法では、次元圧縮の方法として概念距離を求め、求めた概念距離をクラスター分析する。そこで文書のデータサイズが大きくなると計算に非常に時間がかかることが分かった。6.4 節のファイルでは 1 時間以上かかる場合があった。

第7章　考察

本手法で次元数を変化させながら類似度の変化を樹形図で確認した。確認した結果、LSIを利用した方法と同じ次元数の場合は類似度計算の結果はあまり良くないが、次元数を増やすことで結果がよくなることが分かった。次元数を増やすことによる分類の結果の中にはLSIを利用した方法よりよく分類できていると考えられるものもあった。LSIを利用した方法と比較をしたところ6.2節と6.3節では、辞書登録されている名詞を使用した方の結果と分類が似ていることが分かった。これは、3.4.1節に述べた、計算に使用できる名詞が少ないことによる影響は大きいと考えられる。

次元数を増やしていくことで、樹形図の高さが増加することと樹形図の変化がほぼなくなるという結果が得られた。高さの増加については、クラスター-文書行列を樹形図として表示するためのクラスター分析で、次元数が増加することによって起きた変化と考えられる。樹形図の変化は、6.2節のフォルダでは、次元数350から次元数750までは高さの変化以外は全くなくなった。次元数800以降の樹形図は、TfIdfをcos類似度で計算した樹形図とあまり変わらなかった。6.3節では、次元数500以降の樹形図はTfIdfをcos類似度で計算した樹形図とあまり変わらなかった。これは、次元数が増えすぎていることによって単語間の類似度が求められなくなっているためと考えられる。6.4節と6.1節では次元数を樹形図の変化がなくなることはなかった。これは、6.4節と6.1節のように多くの文書では、単語のクラスターへの分類が変わることによる影響が大きいためと考えられる。

6.2節の結果から、6.3節と6.4節のフォルダで次元数が500までの間を100ずつ調べた、結果として次元数100程度まで増やすことで類似度計算の結果が明らかに良くなることがわかった。本手法で使用する次元数については、次元数を100次元程度に単語をまとめることで文書が分類できるようになるのではと考えられる。

第8章　結論

本研究では、Web シラバスから入手した約 3 万教科分のシラバスを対象とし、学科別と高専別に結合したテキストを用いて文書間の類似度計算を行った。類似度計算の手法には、日本語 WordNet を用いた名詞間の概念距離を利用し、次元圧縮する類似度計算を提案した。単語の重要度として TfIdf を使用した。

次元圧縮を累積寄与率 80%となる次元数では、類似度計算の結果が良くなかった。そこで、クラスターに分類された単語と分類された単語に共通する synset について確認を行った。その結果、累積寄与率 80%となる次元では、単語間の類似性は低く、単語に共通する synset は抽象的なものが多いことが分かった。

次元数を変化させることで、単語間の類似度が高くなり、本手法ではよく類似度計算できるのではと考え実験を行なった。比較対象としては、LSI を利用した類似度計算を使用した。文書中の全名詞と辞書登録されている名詞のそれぞれで LSI を利用した類似度計算を行い、その樹形図を比較対象とした。辞書にない名詞の影響によって類似度計算の結果が変化することが分かった。次元数を増やすことで、本手法では類似度計算の性能が上がった。しかし、次元数を増やし続ければ結果が良くなるわけではなかった。

LSI と比較した結果、本手法の方がより良い分類ができる場合があった。また、文書中の全て名詞を利用した場合より辞書にある名詞で LSI を利用した類似度計算を行った樹形図の方が、本手法と結果が近くなった。

単語間の概念距離を利用したクラスター分析による次元圧縮には、100 程度に単語を分類することでよい類似度が求められるであろうと判断した。

謝辞

最後に、本研究を進めるにあたり、ご多忙中にも関わらず多大なご指導を賜わりました出口利憲先生に深く感謝するとともに、 同研究室において共に勉学に励んだ皆様に厚く御礼を申し上げます。

参考文献

[1] 服部 修平, テキストマイニングによる文書の類似度計算に関する研究, 岐阜工業高等専門学校電気情報工学科卒業研究発表 (2017)

[2] 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).

[3] 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).

[4] 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.

[5] 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)

[6] 服部 修平, 田島 孝治, 出口 利憲, 概念距離を利用したクラスター分析による次元圧縮, 平成 29 年度 電気・電子・情報関係学会 東海支部連合大会 (2017)