原著論文

講義データベース—オントロジーを用いた要求知識と習得知識の記述による講義情報の構造化—†

関谷 貴之*・山口 和紀*

学術領域の細分化によって、似た内容の講義が複数の学部や学科で行われる事は珍しくない。多くの組織はシラバスを公開しているが、異なる学部や学科のシラバスを横断的に検索する手段は少ない。そのため、学生が学びたい内容の講義を探したり、教員が自身の講義内容を検討する上で関連のある講義を探したりすることは困難である。この問題を解決するために、我々は大学内でシラバスを検索可能な「講義データベース」を構築した。講義データベースでは講義間の関係を調べるために、共通のオントロジーを用いて、講義の履修に必要な要求知識と受講する事で得られる習得知識を定義した。更に、オントロジーの構造を用いる事で、知識記述における表記の揺れによる影響を緩和可能な講義間の関連度の計算式を導入した。本論文では、講義情報の記述方法とオントロジーの構造、関連度の計算式を説明する。更に、講義データベースのプロトタイプと、講義データベース上に、実際に講義の登録や検索などを行った検証結果について報告する。

キーワード：シラバス、講義データベース、オントロジー、要求知識、習得知識

1. はじめに

大学において、知識や情報を学生に提供することは重要な仕事の一つである。教育は講義という形で提供されており、学生にとっては、学びたい内容の講義がどこで開かれているかを知ることは重要である。また、教員にとっても、例えば自身が担当する講義の範囲や難易度などを検討する上で、関連のある講義や学生が事前に履修している講義の内容を知る必要がある。

講義の内容の概略であるシラバスを用意することは既に広く行われている。しかし、領域の細分化が進む事で、似た内容の講義が複数の組織で行われる事も珍しくないのに対して、本学においては、学部や学科などの組織毎にシラバスが提供されており、異なる学部や学科のシラバスを横断的に検索する手段は用意されていない。そのため、どこでどのような分野の講義が開かれているかを知ることは容易でない。

この問題を解決するためには、分散したシラバスの情報を集約し、共通の概念を用いて記述しなおし、その情報を効果的に検索可能な仕組みを構築する必要がある。本論文ではその仕組みを提案する。まず、共通

の概念としてのオントロジーを用いて、講義の履修に必要な「要求知識」と講義を受講することで得られる「習得知識」を記述する。そして、オントロジーの構造を用いる事で、知識記述における表記の揺れによる影響を緩和可能な講義間の関連度の計算式を導入した。このオントロジーを用いた知識記述と講義間の関連度の式を用いて、東京大学内で互いに関連のある講義の情報を検索可能な「講義データベース」を試作した。さらに、講義データベースの有効性を検証するために、本学で開講されている情報学関連の講義の情報を収集・登録して理解しやすい形で整理し、関連する講義の検索などを行った。本論文ではその手法と検証結果について報告する。

2. 関連研究

従来冊子形式などで提供されたシラバスを WWW 上で公開するオンラインシラバスのシステムは、非常に多くの組織で実現されている。例えば佐賀大学[1]では、全学の講義情報を登録したデータベースを公開している。他にも北大、広島大学など多くの大学で同様のシステムが構築されている。このようなオンラインシラバスでは、講義のタイトルや講義の担当教員などの基本的な情報、受講対象の学生、単位数などの学務上の情報の他、講義の内容を表す情報として、例えば表1のような情報が提供されている。

オンラインシラバスには、従来冊子など紙ベースで提供されたデータを WWW に載せただけのものも多

† Course Database—Systematizing Courses Using Ontology Based on the Descriptions of Prerequisite and Acquired Knowledge — Takayuki SEKIYA and Kazunori YAMAGUCHI

* 東京大学情報基盤センター情報メディア教育研究部門
Campuswide Computing and Communications, Information Technology Center, the University of Tokyo

表1 シラバスの項目例

項 目	内 容
目標	講義の目標
計画	講義の進め方など
履修条件	履修に必要な条件
キーワード	講義の内容を端的に表す語彙
分類	講義形式(座学・実習など)や分野(数学・物理など)、程度(基礎・応用・総合など)など

いが、電子化する事で付加的な機能を付けたものもある。

例えば、東京電機大学では、学生が履修計画を作る事ができるダイナミックシラバス[2]を運用している。ダイナミックシラバスでは、学生が履修したい講義を選択すると、それを履修するために必要なその他の講義を表示するなどして、学生が容易に履修計画を立てることができるような機能を提供している。

名古屋大学高等教育研究センターは、教員のコースデザインを支援する仕組みとしてゴーイングシラバスを開発している[3]。

大学評価・学位授与機構の井田らは、情報処理技術を活用した大学情報(特に教育に関する情報)の収集とデータベース化および情報構造の解析の検討を進め評価等の支援システム構築に関する研究を行っており、その中で各大学のシラバスを集めて分類・整理している[4]。

The University of Texas の World Lecture Hall[5]は、シラバスも含め、講義資料などをインターネット上で共有する場を提供している。World Lecture Hallは、講義資料を提供するサイトやWWWページに限定したディレクトリを提供しているといえよう。

以上の研究は冊子体のシラバスの拡張や応用であり、本研究とは方向性が異なるものである。

東京大学工学部教育プロジェクト室では、学部内の講義の関係を可視化する試みを行っている[6]。具体的な目的を分類の軸として、三次元空間内に各講義を配置して可視化する試みを行っている。

University of Crete の Kotzions らは、講義間の関係やその講義で扱う項目間の関係を定義し、学習者が受講する講義を決めるのを支援するポータルサイトを構築している[7]。この研究では、本研究で扱う講義に関する情報と同様のデータを扱っているが、本研究で対象とするオントロジーの構築やオントロジーを用いた関連する講義の検索については扱っていない。

上記のように、先行研究は講義内容にそって講義同士の関連などを表現するもので、オントロジーを活用

したものはない。しかし、実際のシラバスでは必ずしも用語が統一されておらず、オントロジーを活用する事でより効果的に検索等が可能になると期待できる。そこで、我々は東京大学のシラバスをオントロジーを活用して構造化する事を試みた。

3. シラバスの構造化

3.1 シラバス情報のデータ形式

本研究では、シラバスの構造化を、講義の情報を理解しやすい形に分類・整理して、講義間の関係、内容の関連などを明らかにするという意味で用いる。講義間の関係を示すために、個々の講義に対して前提となる講義を定義したのでは、新たに講義を追加するたびに既存の講義と比較する必要がある。そこで、講義の情報を、そこから関係が抽出できるような形で記述することを考える。そのため、講義の情報として、「カテゴリ」と「要求知識・習得知識」を導入する。

3.2 カテゴリ

図書の十進分類表やWWWの検索エンジンのディレクトリなど、対象を要素の集合に分けて、各集合に適当なラベルをつけることは、広く一般に行われている。そこで、講義を分類するための情報として、カテゴリを導入する。

本研究では、日本図書館協会が出版し、図書館の資料の分類などに用いられている日本十進分類表[8]などを参考にして、階層的なカテゴリを定義した。但し、階層的といっても排他的な分類ではなく、各講義は複数のカテゴリに分類されても良いものとする。表2に本研究で定義したカテゴリの例、図1に、本研究で定義したカテゴリの階層の一部を示す。

当初のカテゴリの階層は2段階であったが、情報学関連の講義の情報を登録する中で、コンピュータやネットワーク等に関わるカテゴリを追加したため、現在4段階となっている。

3.3 要求知識・習得知識

学部や学科が配布する時間割表などでは、「講義 L_A は、講義 L_B を受講済、或は相当の知識を有すること

表2 カテゴリの例

項 目	定 義
名前	メカトロニクス
上位カテゴリ	機械工学, 電子工学
コメント	—

図1 カテゴリーの階層の一部

表3 知識の例

項 目	定 義
名前	木
カテゴリ	データ構造
上位知識	データ構造
全体知識	—
説明	計算機上のデータモデルの一つ

を履修条件とする」といった記述が見られる。これは、各講義について、講義を受講するために要求される知識、また受講することで習得可能な知識が存在し、かつこれをシラバス作成の際に考慮していると考えられる。

上記の講義に関する知識を正確に定義するには、知識の対象領域や学生の理解度など考慮すべき点が多いが、本研究では個々の知識の具体的な内容には踏み込まず、「情報処理」のような記号で表すこととする。

本研究において個々の知識は、名前、その知識が属するカテゴリ、その知識の説明、a kind of 関係に相当する上位知識、a part of 関係に相当する全体知識で定義する。表3に知識の例を示す。

このように知識を定義することで、関係付けられた知識の体系、ならびにこの知識を用いて記述した講義の履修に必要な要求知識と受講することで得られる習得知識の情報が、図2に示すようなシラバスにおけるオントロジー[9]を形成する。

図2 シラバスにおけるオントロジーの一部

表4 講義の例

項 目	定 義
タイトル	計算機プログラミングⅠ
内容	情報処理は現代社会の基本的な枠組となりつつある。基礎科目「情報処理」ではその基礎的な操作や…
講師	山口 和紀 教授
カテゴリ	プログラミング
要求知識	情報処理、情報リテラシー
習得知識	アルゴリズム、データ構造 プログラミング

図3 講義と知識の関係

3.4 講義

前述の「カテゴリ」と「要求知識・習得知識」を用いて、講義を記述した例を表4に示す。この情報から知識と講義をノード、その間の関係をアークとしてグラフを作ることができる。例を図3に示す。

図3から以下の関係が読み取れる。

- (1) 講義「情報処理」を受講すると「情報処理」の知識が習得できる。
- (2) 講義「計算機プログラミングⅠ」の履修には「情報処理」の知識が要求される。
- (3) 講義「計算機プログラミングⅠ」又は、講義「コンピュータプログラミング」を受講すると、「プログラミング」の知識が習得できる。

これによって、ある講義の履修に必要なと思われる講義が分かる、つまり講義間の先行関係を示すことができる。

3.5 オントロジーに基づく講義間の関係

本研究では講義の要求知識や習得知識の関連から、講義自体の関連がある程度求められると仮定する。

講義間の関連性の算出には、ベクトル空間モデル[10]の処理方法の一つである *tf·idf* 法を、オントロジーを考慮して改良したものを用いる。ベクトル空間モデルは、文書やWWWページなどテキスト間の関連性の度合いを測るために、広く用いられる手法である。

具体的には、それぞれの講義の要求知識や習得知識から、3.3節で述べたオントロジーを用いて、「要求知識ベクトル」と「習得知識ベクトル」を作る。これらのベクトルは、以下のように算出する。

講義 L の要求知識ベクトル $PV(L)$ を(1)式で定義する。ここで、 $C_1, C_2, \dots, C_n$ は知識を全て列挙したものととする。

$$PV(L) = (pv(L, C_1), \dots, pv(L, C_n)) \quad (1)$$

$pv(L, C_k)$ は、講義 L と知識 C_k の関連度であり、講義 L の要求知識 $Cpv(L) = \{C_x, C_y, \dots\}$ に対して、(2)式で定義する。

$$pv(L, C_k) = tfpv(L) \cdot (rel(C_x, C_k) \cdot idfpv(C_x) + rel(C_y, C_k) \cdot idfpv(C_y) + \dots) \quad (2)$$

$tfpv(L)$, $idfpv(C)$ は、それぞれ *tf·idf* 法における TF (Term Frequency) と IDF (Inverse Document Frequency) に相当するものである。 $tfpv(L)$ は(3)式で定義する。ここで $numpv(L)$ は、講義 L の要求知識の数である。

$$tfpv(L) = \frac{1}{numpv(L)} \quad (3)$$

多数の要求知識を定義した場合は、講義全体の中で個々の知識に関わる内容は少なくなると考えられるので、一つ一つの知識の重要性を小さくするために(3)式を採用した。

$idfpv(C)$ は、(4)式で定義する。ここで、 L_{all} は全講義数であり、 $numcoursepv(C)$ は、知識 C を要求知識として持つ講義の数である。

$$idfpv(C) = \log\left(\frac{L_{all}}{1 + numcoursepv(C)}\right) \quad (4)$$

$rel(C_x, C_k)$ は、知識 C_x と知識 C_k の関連度を示す値である。知識のオントロジー(3.3節参照)から、知識をノード、知識間の関係をアークとする有向グラフを作る。図4に例を示す。この時、 $rel(C_x, C_k)$ を(5)式で

図4 知識間の関係

定義する。ここで、 $C_x, C_p, C_q, \dots, C_r, C_k$ を、ノードとしての知識 C_x から知識 C_k との間のパスとする。

$$\begin{aligned} rel(C_x, C_k) &= \max(arc(C_x, C_p) \cdot arc(C_p, C_q) \cdot \dots \\ &\quad \dots \cdot arc(C_r, C_k)) \end{aligned} \quad (5)$$

$\max()$ は、 C_x から C_k への全てのパス $C_x, C_p, C_q, \dots, C_r, C_k$ に対する $arc(C_x, C_p) \cdot arc(C_p, C_q) \cdot \dots \cdot arc(C_r, C_k)$ の最大を取る。

$arc(C_x, C_p)$ は、知識 C_x と C_p との関係から決まる知識間の関連度を表す値である。知識 C_p が知識 C_x の下位知識で、 C_x の下位知識の個数を $numarc(C_x, C_p)$ とした時に、 C_x の下位知識が多い場合は、 C_p と C_x との関わりは小さくなるが、 $1/numarc(C_x, C_p)$ 程には小さくならないものと仮定する。そこで、対数を使うこととして $arc(C_x, C_p)$ を(6)式で定義する。ここで W_A は知識間の関係の種類に応じて決まる定数とする。

$$arc(C_x, C_p) = \frac{W_A}{\log(1 + numarc(C_x, C_p))} \quad (6)$$

W_A は、直接の関係がない知識 C_x, C_k の $rel(C_x, C_k)$ が、直接の関係がある知識同士 C_x, C_p の $rel(C_x, C_p)$ よりも小さくなるように決定することにする。従って、互いに関係のある知識 C_a, C_b に対して、(7)式が成り立つ必要がある。

$$arc(C_a, C_b) < 1 \quad (7)$$

以上で定義した要求知識ベクトル $PV(L)$ と同様に、講義 L の習得知識ベクトル $AV(L)$ を(8)式で定義する。 $C_1, C_2, \dots, C_n$ は知識を全て列挙したものである。

$$AV(L) = (av(L, C_1), \dots, av(L, C_n)) \quad (8)$$

講義 L の習得知識 $C_{av}(L) = \{C_x, C_y, \dots\}$ に対し

て, 講義 L と知識 C_k の関連度 $av(L, C_k)$ を (9) 式で定義する.

$$\begin{aligned} av(L, C_k) &= tfav(L) \cdot (rel(C_x, C_k) \cdot idfav(C_x) + \\ &\quad rel(C_y, C_k) \cdot idfav(C_y) + \dots) \end{aligned} \quad (9)$$

$tfav(L)$, $idfav(C)$ は, それぞれ $tf \cdot idf$ 法における TF(Term Frequency) と IDF(Inverse Document Frequency) に相当するものである. $tfav(L)$ は, 講義 L の習得知識の数 $numav(L)$ に対して, (10) 式で定義する.

$$tfav(L) = \frac{1}{numav(L)} \quad (10)$$

$idfav(C)$ は, 知識 C を習得知識とする講義の数 $numcourseav(C)$ と, 全講義数 L_{all} に対して, (11) 式で定義する.

$$idfav(C) = \log\left(\frac{L_{all}}{1 + numcourseav(C)}\right) \quad (11)$$

要求知識ベクトルと習得知識ベクトルのそれぞれの内積を用いて, 講義 L_A と L_B の関連度 $Sim(L_A, L_B)$ を (12) 式のように定義する.

$$\begin{aligned} Sim(L_A, L_B) &= \frac{PV(L_A) \cdot PV(L_B)}{|PV(L_A)| |PV(L_B)|} \cdot (1 - W) + \\ &\quad \frac{AV(L_A) \cdot AV(L_B)}{|AV(L_A)| |AV(L_B)|} \cdot W \end{aligned} \quad (12)$$

W は関連度において, 習得知識と要求知識のどちらを重視するかを表わす重みである. 通常, 要求知識 (その講義を受講するための要件となる知識) が異なっても, 習得知識 (その講義を受講することで学ぶことのできる知識) が似ていれば, その講義は似ていると考えることが多いので, 重みを $0.5 < W < 1$ となるように定めることにする.

本研究では, この関連度 $Sim(L_A, L_B)$ が 0 より大きければ, 講義 L_A と L_B には関連があると考え, 値が 1 に近いほど関連が強いとする.

4. 講義データベース

4.1 システム構成

講義データベースは, 各講義に関する情報を登録したデータベースと, その講義情報の参照や管理を行う WWW インタフェースとしての Perl CGI, ならびに

表 5 知識間の関係

	l	n	l/n
上位関係	163	153	1.065
下位関係	163	72	2.264
全体関係	116	115	1.009
部分関係	116	47	2.468

関連講義計算のための前処理などを行う Perl のプログラム等で構成される.

現在, 約 650 の講義の情報を 430 個の知識と 113 個のカテゴリを用いて記述している [11].

4.2 関連講義の提示

データベース上の講義情報を参照する CGI では, 指定した講義から (12) 式で求めた関連度の高い講義を表示する. 計算に当たっては, 各式の定数等を次のように設定している.

(6) 式の W_A については, オントロジーの構造を考慮する必要がある. 例えば, ある知識が要求知識である場合に, その知識の下位の知識が多く上位の知識が少ない場合, W_A を同じ値にすると, 上位知識が下位知識よりも要求知識ベクトルに大きく寄与する. 知識間の関係の意味や, それが講義の特徴づけにどれだけ効いてくるかを一般的に決定することは容易ではないが, 本研究では両者が同程度に影響を及ぼすものと仮定する.

本研究において, オントロジーの全体構造に制約を課していないが, 現在講義データベースに蓄積された知識では, ある知識から見て上位関係にある知識は, 下位関係にある知識よりも少ない傾向がある. 表 5 に, 上位下位などの関係の数 l をその関係を持つ知識の数 n で割って求めた. 各知識に対して上位下位などの関係にある知識の数の平均を示すものである.

直接関係のある知識 C_x と C_k について, 関係の度合い $rel(C_x, C_k)$ が, 関係の種類によらず同程度になるものとする. (13) 式が成り立つ. ここで, W_{Asuper} , W_{Asub} , W_{Awhole} , W_{Apart} は, 上位, 下位, 全体, 部分のそれぞれの関係にある場合の, (6) 式の定数 W_A に相当する.

$$\begin{aligned} \frac{W_{Asuper}}{\log(1 + 1.065)} &= \frac{W_{Asub}}{\log(1 + 2.264)} = \\ \frac{W_{Awhole}}{\log(1 + 1.009)} &= \frac{W_{Apart}}{\log(1 + 2.468)} \end{aligned} \quad (13)$$

要求知識や習得知識として定義した知識 C_x が, 直接関係のある知識 C_p に及ぼす影響は, 平均して 0.5 であると仮定すると, (13) 式より, 次のように定数

W_{Apart} の値が求められる.

$$\begin{aligned}\frac{W_{Apart}}{\log(1+2.468)} &= 0.5 \\ W_{Apart} &= 0.5 \log(1+2.468) \\ &\simeq 0.622\end{aligned}\quad (14)$$

同様に求めた他の定数を, 表 6 に示す.

この値は, 以下に示すように (7) 式を満たす.

$$\frac{W_{Apart}}{\log(1+1)} \simeq \frac{0.622}{0.693} < 1 \quad (15)$$

また (12) 式における習得知識に関する重み W については, 3.5 節で述べた W ($0.5 < W < 1$) を満たす中で, 習得知識にやや重みをおくこととして, 本研究では $W=0.7$ と設定した. なお, 関連度は重みに関しては依存性が低く, この数値を多少変動しても以下の議論が大きく変わるものではない.

計算時間の都合から, (5) 式については, オントロジーのグラフ表現において, 要求知識や習得知識 C_x からのパスの長さが 2 以下の範囲で計算する. つま

表 6 知識間の関係に関わる重み

知識間の関係	定数名	値
上位関係	W_{Asuper}	0.362
下位関係	W_{Asub}	0.591
全体関係	W_{Awhole}	0.348
部分関係	W_{Apart}	0.622

り, (16) 式を用いる.

$$\begin{aligned}rel(C_x, C_k) &= \begin{cases} 1 & a = 0 \\ arc(C_x, C_k) & a = 1 \\ arc(C_x, C_p) \cdot arc(C_p, C_k) & a = 2 \\ 0 & a > 2 \end{cases} \quad (16)\end{aligned}$$

これは, 異なる知識の関連度については上位の知識と同じもののだけを考慮することを意味する.

以上の式と値を用いて, 講義データベースでは, 指定された講義と関連度が高い講義を表示する. 図 5 は, 講義データベースの「計算法プログラミング I」の表示である. 画面下部に「関連講義」として, 関連度の高い講義の名称と関連度を表示している.

[タイトル]

計算法プログラミング II CMap www

[内容]

計算法プログラミング II

この講義では, グラフ, リスト, 集合, 木などの基本的なデータモデルについて学ぶ.

受講対象者は, 基礎科目「情報処理」履修済みに相当する人で, プログラミング経験のある人とする.

プログラミング経験は <http://hw.b.ecc.u-tokyo.ac.jp/topics/programming/index.html> を参考に, 目い.

講義に関連する情報は <http://lecture.ecc.u-tokyo.ac.jp/~yamaguch/index.html> で提供している.

質問は yamaguch@ecc.u-tokyo.ac.jp の電子メールで受け付ける.

[講師]

山口 和紀 教授

[カテゴリ]

情報科学, データ構造

[要求知識]

情報処理, 情報リテラシー, プログラミング

[習得知識]

グラフ, データ構造

[関連講義] 迄

- 算法数理工学(0.547370819434282)
- 計算法プログラミング(0.530722777603022)
- 計算法プログラミング(0.530722777603022)
- グラフ分割アルゴリズム(アルゴリズム設計論)(0.494974746830583)
- 情報数学(0.494974746830583)
- 離散構造とアルゴリズム(0.494974746830583)
- コンピュータアルゴリズム(0.442718872423573)
- ソフトウェア第三(0.369961860554451)
- プロセスシステム工学及び演習(0.35)
- システム環境 II(0.285773803324704)

図 5 関連講義

図6 講義と知識, その間の関係の可視化

4.3 可視化

講義と知識の関係, 講義同士の先行関係などを表示する CGI は, 知識や講義をノードとして, 要求知識や習得知識による知識と講義の関係をアークとして図示する. 図6は, 「計算機プログラミングI」を中心とした表示例である. この表示では, 背景が白のノードとして講義を, 背景が黄緑色のノードとして知識を表示している. 講義と要求知識とは赤いアークで, 習得知識とは青いアークで結んでいる. また黄色いアークは知識間に上位・下位関係又は部分・全体関係があることを示している. 個々のノードをクリックすることで, ノードに対応する講義や知識の定義を確認することができる. また, 表示された知識を, 要求知識や習得知識として持つ他の講義を表示することもできる.

4.4 講義情報の入力支援

現在講義データベースに登録されているデータのほとんどは, 登録済みの講義や知識のデータを適宜参照しながら, 著者らが自ら登録したものである.

実際の利用を考えると, 講義を行う教員やシラバスのデータを扱う職員等が, 容易に入力できなければならない. そこで, そのようなデータ入力を支援する仕組みを現在開発中である. 図7は, 講義内容の文章から, 要求知識や習得知識の候補を提示するインタフェースである. このインタフェースの仕組みは次の通りである.

- (1) 利用者が講義の内容を WWW インタフェースから入力する.
- (2) 講義データベースが, 登録済みの講義の内容と入力された講義の内容を比較して類似度を算出する. 算出には, 汎用連想計算エンジン GETA[12]を用いる. GETAは頻度付きの索引データ等に基づいて内積型の類似度を高速に計算するツールである. ここでは GETA を用いた関連文書検索の機

能^{*}を用い, 文章内の単語の出現頻度に基づいて類似度を算出する.

- (3) 個々の知識に対して, その知識を要求知識や習得知識として持つ各講義の(2)で求めた類似度を積算する.
- (4) (3)で求めた値が高い知識を, (1)で入力された講義の要求知識や習得知識の候補として示す.

図7では, 上記の(1)で入力した文章を講義内容としてインタフェースの上部に, 上記の(4)で述べた要求知識や習得知識の候補を, 講義内容の下に表示している.

5. 考察

5.1 講義データベースの効用

本研究の一つの成果としては, 講義データベースのようなシラバスのデータベースを, 組織の枠を越えて提供できた事自体である. 現在は, シラバスの入手しやすさなどから, 主にコンピュータやプログラミングに関する講義の情報を登録しているが, 例えば「プログラミング」に関する20以上の講義が, 15以上の学科で開かれていることが分かった.

5.2 講義間の関連度に対するオントロジーの影響

本研究では, オントロジーを考慮して講義間の関連度を計算しているが, 考慮する度合いを変化させる事で, オントロジーをどのように用いるべきかについて考察する.

具体的には, 4.2節において仮定した, 直接関係のある知識間の影響を示す値 0.5 を, 0.05 から 0.95 まで 0.05 刻みで変化させ, ある講義と関連のある講義との関連度 *Sim* の変化を調べた. 以下では, この知識

^{*} 詳しくは汎用連想検索エンジン (第3版) 導入・操作マニュアルの「連想ライブラリ (libae) のチュートリアル」を参照のこと.

表7 講義データベース内の講義の一部

講義 ID	講 義 名
7	計算機プログラミング I
8	計算機プログラミング II
27	計算機プログラミング I
53	システム数理実習
57	情報システム科学 V
60	情報システム科学実習 II
199	数理情報学 I 演習
208	言語情報解析実験実習 II
221	アルゴリズム設計
222	シミュレーション学
225	ビジュアルコンピューティング論
240	人工環境学基礎演習
257	アルゴリズム論
276	コンピュータ科学特別講義 I
281	プログラム構造論
347	図形科学実習
384	数理手法 III
399	ソフトウェア第一
404	数値熱流体工学
421	計算機工学概論
421	計算機工学概論
427	コンピュータアルゴリズム
443	電気電子情報実験第二
445	計算機科学概論
449	算法言語
461	コンピュータおよび演習
463	コンピュータ化学演習
469	ツールとしての計算機 2
489	プログラミング基礎 D
498	離散構造とアルゴリズム
533	アルゴリズムとデータ構造

間の影響の度合いを「知識間関連度」、講義間の関連度 Sim を「講義間関連度」と呼ぶこととする。

図8は、講義データベース中の講義「計算機プログラミング I」と他の講義との講義間関連度の変化を表わしている。縦軸は講義間関連度、横軸は知識間関連度である。グラフの右に凡例として示した番号は、講義データベース内の講義につけられた ID である。個々の ID が示す講義の名称を表7に示す^{☆☆}。図8には、計算機プログラミング I との講義間関連度が高い講義のみ示している。

☆☆ 講義データベース[11]にアクセスすることで、個々の講義の定義の詳細を確認できる。

どのグラフも知識間関連度が大きくなるに従って、講義間関連度が大きくなっている。これは計算機プログラミング I と関連がある講義の双方で、要求知識や習得知識として互いに関連のある知識を定義している事から、知識間関連度が大きくなるに従って、(12)式で用いるベクトル PV , AV でのそれらの各知識に対応する値が大きくなって、ベクトルの内積が長さに対して大きくなり、講義間関連度が高くなるからである。

一方、講義「システム数理実習」と他の講義との講義間関連度の変化を示したのが、図9である。ここでは、計算機プログラミング I とは反対に、知識間関連度が大きくなるに従って、講義間関連度が小さくなっている。これは、システム数理実習の要求知識の知識「数値計算」と関連する知識が、講義データベース中に登録されておらず、かつシステム数理実習と関連がある講義では、他に多くの要求知識を持つ事から、知識間関連度が大きくなるに従って、(12)式で用いるベクトル PV , AV でのシステム数理実習とは関係のない知識に対応する値が大きくなって、ベクトルの長さが内積に対して大きくなり、講義間関連度が低くなるからである。

このように、関連度の算出対象となる講義の要求知識や習得知識に関連する知識が定義されているかどうかというオントロジーの構造によって、知識間関連度の増加は、講義間関連度を増やす方向にも減らす方向にも働く場合があることが明らかになった。

講義データベース上のオントロジーを構築する著者らの立場では、知識間関連度の値を変えられるような仕組みを作成すれば、それによって変化する講義間関連度の高い講義を見ることで、講義と知識の関係を知るための参考として使える。一方、同じ仕組みを講義データベースの利用者にも提供すれば、関連する講義として利用者が納得できる講義が見つからなかった場合に、範囲を広げて探すことができるようになる。

5.3 オントロジーの利用と構築

5.2節で示した知識間関連度と講義間関連度との比較から、本研究における講義間の関連度の算出方法を用いる場合は、関連する知識が複数定義された知識を要求知識か習得知識として定義した方が、関連度の高い講義を多数見つける事ができる。これは講義情報を定義する時の指針となるだろう。

オントロジーを考慮して講義間の関連性を示すことを通して、オントロジーとしての知識間の関係を確認することができる。例えば、ある講義と関係度の高い講義として、明らかに関係がない講義が示された場合

は, それらの講義の要求知識が習得知識, 更にはそこで定義された知識と上位下位や部分全体関係にある知識に不適切なものがあるかも知れないと推測できる。

6. 終わりに

本論文では, 我々が構築している講義データベースにおいて, 講義同士の関係を提示できる仕組みを報告した。これによって, シラバスを構造化して, 教員や学生が知りたい内容の講義や, 関連のある講義を提示することが可能になった。オントロジーの構造や知識間関連度などのオントロジーを考慮する度合いによって, 提案した講義間の関連度がどのように変化するかを明らかになった。式自体の妥当性の検証や重みなどのパラメータの調整, カテゴリの活用などは今後の課

題である。

本研究では, 適切なオントロジーが存在し, それに応じた講義情報の登録ができることを仮定しているが, 実際にそれが可能であるためには, 登録する人を支援する仕組みが必要である。講義データベースには, 要求知識と習得知識の候補を提示するなどの機能があるが, 今後更に知識間関連度を調整して講義間関連度が変化する模様を示すことのできるインタフェースを構築していきたい。これを使う事で教員や学生が関連する講義をより幅広く探したり, 講義情報の定義やオントロジー構築の参考にしたりすることができるようになる。このようにオントロジーの構築を効果的に支援する方法や, ある程度自動的に構築する方法について, 今後研究を進めていく予定である。

The screenshot shows a web application titled "Course DB (講義データベース)". It displays the "講義内容" (Course Content) for a course on "情報処理" (Information Processing). The content describes the course's focus on basic operations and concepts, including algorithms, data structures, and file processing. Below the content, there are two sections: "要求知識の候補" (Candidate Required Knowledge) and "習得知識の候補" (Candidate Acquired Knowledge). Each section lists various subjects and their corresponding numerical values.

要求知識の候補	
情報リテラシー	1.62026384813484
情報処理	1.47354075593176
プログラミング	1.15398798392999

習得知識の候補	
情報システム	3.48264989172343
数値解析	3.16772659925336
CAD	3.03327291444292
数値計算	2.73645352522459
データ解析	2.71018555497026
情報リテラシー	2.68275457811351
設計論	2.23317168053682
遺伝的アルゴリズム	2.10873811581487
最適化	2.06479270320774
Webページ	1.96525848207707
並列アーキテクチャ	1.8865798640643
文字コード	1.87948076368127
WWW	1.85155064426695
ファジー	1.83565166212699
ニューラルネットワーク	1.83090694653963

At the bottom of the interface, there is a URL: http://www.tokyo.ac.jp/coursedb/concept_info.cgi?id=214 and a note: "を新ウィンドウで開く" (Open in new window).

図7 入力支援：要求知識と習得知識の候補の提示

図8 「計算機プログラミングI」と他の講義との講義間関連速度

図9 「システム数理実習」と他の講義との講義間関連速度

参 考 文 献

- [1] 佐賀大学：佐賀大学オンラインシラバス (1998) . <http://syllabus.dl.saga-u.ac.jp/>.
- [2] 土肥紳一, 中村尚五, 島田尊正, 川勝真喜：導入教育におけるダイナミックシラバスの活用, 平成13年度情報処理教育研究集会, pp.275-278 (2001) .
- [3] 池田輝政, 戸田山和久, 近田政博, 中井俊樹：成長するティップス先生—授業デザインのための秘訣集, 玉川大学出版部 (2001) .
- [4] 井田正明, 野澤孝之, 芳鐘冬樹, 宮崎和光, 喜多一：

シラバスデータベースシステムの構築と専門教育課程の比較分析への応用, 大学評価・学位研究2, 大学評価・学位授与機構 (2005) .

- [5] WHL administrator : World Lecture Hall (2004) . <http://web.austin.utexas.edu/wlh/>.
- [6] 東京大学工学部教育プロジェクト室：教育プロジェクト室の活動と重点課題 (2004) . <http://www.t.u-tokyo.ac.jp/tpage/esp/gaiyo.html>.
- [7] Kotzinos, D., Pediaditaki, S., Apostolos, A., Athanasios, N. and Christophides, V. : Online Curriculum on the Semantic Web : The CSD-UoC Portal for

- Peer-to-Peer E-learning, *The 14th International World Wide Web Conference(WWW2005)*, pp.307-314 (2005) .
- [8] 日本図書館協会：日本十進分類法(NDC)新訂9版(1995) .
- [9] 溝口理一郎：オントロジー研究の基礎と応用，人工知能学会誌，Vol.14，No.6，pp.977-988 (1999) .
- [10] Salton, G. and McGill, M. : *Introduction to Modern Information Retrieval*, McGraw-Hill, New York, NY (1983) .
- [11] 東京大学情報基盤センター：講義データベース(2005) . <http://coursedb.its.u-tokyo.ac.jp/>.
- [12] 国立情報学研究所：汎用連想計算エンジン (GETA) (2003) . <http://geta.ex.nii.ac.jp/>.
(2005年3月31日 受付)
(2005年8月5日 採録)
- [問い合わせ先]
〒153-8902 東京都目黒区駒場3-8-1 東大駒場第一キャンパス情報教育棟内
東京大学情報基盤センター情報メディア教育研究部門
関谷 貴之
TEL : 03-5841-3009
FAX : 03-5454-4346
E-mail : sekiya@ecc.u-tokyo.ac.jp

 著者紹介

 せきや たかゆき
関谷 貴之 [非会員]

1995年東京大学工学部精密機械工学科卒。1997年東京大学大学院工学系研究科精密機械工学専攻修士課程修了。2000年東京大学大学院工学系研究科精密機械工学専攻博士課程単位取得退学。同年東京大学情報基盤センター助手、博士(工学)、教育支援システムの開発・運用、研究に従事。

 やまぐち かすのり
山口 和紀 [非会員]

1979年東京大学理学部数学科卒。1981年東京大学理学部助手。1985年理学博士(東京大学)。1989年筑波大学電子情報工学系講師。1992年東京大学教養学部助教授。1999年東京大学情報基盤センター教授。コンピュータのためのモデリング全般に興味を持つ。

Course Database – Systematizing Courses Using Ontology Based on the Descriptions of Prerequisite and Acquired Knowledge –

by

Takayuki SEKIYA and Kazunori YAMAGUCHI

Abstract :

Because academic fields are becoming increasingly segmented, various faculties here and elsewhere at a university often offer similar courses. Faculties publish syllabus information about their courses, but there are no means to search for this information across faculties. Consequently, it is not possible for students to conduct university-wide searches when they want to attend the most suitable courses outside their faculties, and difficult for faculty members to conduct systematic searches when they want to refer to other courses related to theirs. This problem motivated us to develop a campus-wide syllabus database, "Course Database." To make it possible to identify relationships between independently registered courses, we associated courses with the descriptions of prerequisite and acquired knowledge using a common ontology. Then, we introduced a similarity measure used for searching the ontology to alleviate differences in terminology used in the various knowledge descriptions. In this paper, we explain the format of the knowledge description, the structure of the ontology, and the similarity measure, introduce our prototype system, and show some experimental results.

Keywords : Syllabus, Course, Database, Ontology, Prerequisite Knowledge, Acquired Knowledge

Contact Address : **Takayuki SEKIYA Kazunori YAMAGUCHI**

153-8902, Japan

TEL : 03-5841-3009

E-mail : sekiya@ecc.u-tokyo.ac.jp