Analysis of Curriculum Structure Based on LDA

Takayuki Sekiya[*], Yoshitatsu Matsuda[†], and Kazunori Yamaguchi[‡]

Abstract— **A good curriculum is crucial for a successful university education. When developing a curriculum, topics, such as natural science, informatics, etc. are set first, and then course syllabi are written accordingly. However, there is no guarantee that the topics actually covered by the course syllabi are identical to the initially set topics. To find out if the actual topics covered by the developed course syllabi, we developed a method of systematically analyzing course syllabi that uses latent Dirichlet allocation (LDA) and Isomap. We applied this method to the syllabi of MIT and the Open University curricula, and verified the method is promising.**

Keywords: Syllabus, Curriculum Analysis, LDA, Isomap

1 Introduction

A curriculum is one of the most important assets of higher education. A well-developed curriculum is essential for effective learning. A curriculum should represent characteristic educational activity that each university provides students with. To design such an original curriculum, faculty has to analyze current curricula given by other universities, however, it is not easy task to grasp characteristics of a curriculum because the analysis of a curriculum requires professional knowledge in various fields.

We are conducting the curriculum analysis. In our previous paper[1], we used syllabi as the target of our research. Assuming that syllabi adequately reflect the contents of a course, we successfully extracted relationships among syllabi utilizing term sets. This relationship is useful for analyzing the local structure of syllabi.

In this paper, we propose a method to generate a map of the syllabi from which we can understand the whole structure of a curriculum represented by its syllabi. To generate such a map, there are two problems. First, the term sets are sparse and the overall structure of the syllabi cannot be determined by term sets. Second, the syllabi are not uniformly distributed in the term vector space.

To overcome the first problem, we use topics instead of term sets. The topics are extracted from syllabi using latent Dirichlet allocation (LDA). To overcome the second problem, we use Isomap[2].

Using these methods, we conducted experiments. First, in order to see the reliability and informativeness of our method, we applied our method to CS2008[3][1], which is the curricular guidance of computer science. Then, we applied our method for comparison of two curricula. We first extracted model topics from CS2008, and created a map of the computer science curricula of MIT as a reference curriculum. Then, we plotted the course syllabi of the Open University (the OU) curriculum into the map of MIT curriculum. From this comparison, we can see how the computer science courses of MIT and the OU cover the standard computer science topics.

In Section 2, basic theories are explained. Experimental results are detailed in Section 3. Previous works are explained in Section 4. Section 5 concludes this paper.

2 Analysis Curriculum Structure

2.1 Criteria of Curriculum Analysis Method

Improper visualization causes misunderstanding and is harmful. To avoid this problem, we first set criteria which a proper visualization should satisfy.

Criterion 1: In order to create a map of a curriculum, we need bases on which courses are represented. Because there are no universal bases, we have to be able to generate bases from a collection of courses.

Criterion 2: The relative position in a map should be correct. In other words, if a course is related to two topics A and B, the course is located between these topics with respect to the meaning of the map.

Criterion 3: In order to compare one curriculum with other curriculum, we need put the two curricula on a same map. It is desirable if we can use one curriculum as a reference and map course syllabi of other curricula on the reference so that other curricula can be compared on the same reference.

2.2 LDA

We use LDA to extract latent topics from course syllabi, and use them as bases required by Criterion 1. LDA is

[*] Information Technology Center, the University of Tokyo, 3-8-1 Komaba, Meguro, Tokyo, Japan 153-8902, Email:sekiya@ecc.u-tokyo.ac.jp

[†] Department of Integrated Information Technology, Aoyama Gakuin University, Email:matsuda@graco.c.u-tokyo.ac.jp

[‡] Graduate School of Arts and Sciences, the University of Tokyo, Email:yamaguch@graco.c.u-tokyo.ac.jp

[1] CS2001 has been in reviewing process since 2008. In this paper, we call this modified version of CS2001 "CS2008."

Table 1: The Body of Knowledge of CS2008

DS	Discrete Structures
PF	Programming Fundamentals
AL	Algorithms and Complexity
AR	Computer Architecture
OS	Operating Systems
NC	Net Centric Computing
PL	Programming Languages
HC	Human-Computer Interaction
GV	Graphics and Visual Computing
IS	Intelligent Systems
IM	Information Management
SP	Social and Professional Issues
SE	Software Engineering
CN	Computational Science

a method used to extract latent topics based on a generative probabilistic model of collections of discrete data, such as text corpora. In a variety of LDA models, we use the model proposed by Blei[4] because we can adjust how strongly courses are related to topics by some parameters so that it satisfies Criterion 2.

Given a document-word matrix of text corpora, the LDA produces a set of topics, where each topic is characterized by a distribution over words. In the LDA, a k-topic LDA model assumes the following generative process of an N-word document:

1. Choose $\theta \sim \mathit{Dirichlet}(\boldsymbol{\alpha})$
2. For each of the N words w_n
   (a) Choose a topic $z_n \sim \mathit{Multinomial}(\theta)$
   (b) Choose a word w_n from $p(w_n|z_n, \beta)$, a multinomial probability conditioned on the topic z_n.

$\boldsymbol{\alpha}$: parameter of Dirichlet distribution. θ: topic mixture. $\beta_{ij} = p(w_n = i|z_n = j)$: probability distributions of words over topic z_n (topic-word matrix). $\boldsymbol{w} = (w_1, \cdots, w_N)$: a document (a sequence of words). $p(\boldsymbol{w}|z_n)$: probability distributions of a document over topic z_n (document-topic matrix).

Using EM algorithm, the LDA performs variational inference of θ and z for a document, and estimation of the topics β. The relationship among courses and topics, which is represented by $p(\boldsymbol{w}|z, \beta)$ in the LDA, depends on $\boldsymbol{\alpha}$. $\boldsymbol{\alpha}$ is regarded as a single parameter α in Blei's model, though $\boldsymbol{\alpha}$ is a k-vector in general. α can be set fixed or be estimated by EM algorithm. If we use smaller α, the LDA relates a course to a few topics strongly while if we use larger α a course is related to various topics. Because the relationship among courses and topics directly determines the structure of a map, it is important to use an appropriate α. In this research, through the iteration of generating map of course syllabi, we experimentally decide α so that courses are not extremely concentrated.

2.3 A Syllabus Map Creation by Isomap

For generating a map, we use $p(\boldsymbol{w}|z)$ as coordinates of a syllabus. Because the number of topics is around 10 in our experiments and higher than 2 or 3, we have to reduce its dimension for visualization. In order to utilize the neighborhood structure of syllabi in the high dimensional structure for visualization, we employ Isomap for the dimension reduction. The Isomap is a method used to connect nearby points to form a manifold in a higher dimensional space and unfold it into a low-dimensional space. Therefore, we can reduce the overlay of clusters to satisfy Criterion 2.

The shape of a map generated by Isomap depends on a distribution of syllabus in an original higher dimensional space. To enable the comparison on the same map, we employ the following procedure.

1. Analyzing the reference curriculum C^{ref} by means of LDA, we get β estimated by EM algorithm, which is a probability distribution of words over topics, and also we get $p^{\text{ref}}(\boldsymbol{w}|z)$, which is a probability distribution of course syllabi of C^{ref} over topic z

2. $\boldsymbol{v}_{LDA}(\boldsymbol{w}^{\text{ref}}) = (p^{\text{ref}}(\boldsymbol{w}^{\text{ref}}|z_1), \cdots, p^{\text{ref}}(\boldsymbol{w}^{\text{ref}}|z_k))$ represents a course syllabus $\boldsymbol{w}^{\text{ref}}$ in the topic space. We reduce its dimension using Isomap (see Equation 1) and generate a map of C^{ref} in the 2D space as follows:

$$v_{iso}(\boldsymbol{w}^{\text{ref}}) = \Pi_{\text{Isomap}}(\boldsymbol{v}_{LDA}(\boldsymbol{w}^{\text{ref}})) \quad (1)$$

   where $\Pi_{\text{Isomap}} : \boldsymbol{R}^k \rightarrow \boldsymbol{R}^2$ is the Isomap projection.

3. Analyzing the test curriculum C^{test} by LDA with β acquired at Step 1, we get $p^{\text{test}}(\boldsymbol{w}|z)$, which is a probability distribution of course syllabi of C^{test} over topic z. From $p^{\text{test}}(\boldsymbol{w}|z)$, we get $\boldsymbol{v}_{LDA}(\boldsymbol{w}^{\text{ref}})$.

4. For each $\boldsymbol{v}_{LDA}(\boldsymbol{w}^{\text{test}})$, pick up K-nearest neighboring syllabi of $\boldsymbol{v}_{LDA}(\boldsymbol{w}_i^{\text{ref}})(1 \leq i \leq K)$ where the distance is given as the Euclidean distance between $\boldsymbol{v}_{LDA}(\boldsymbol{w}^{\text{test}})$ and $\boldsymbol{v}_{LDA}(\boldsymbol{w}_i^{\text{ref}})$.

5. Calculate a projection of $\boldsymbol{v}_{LDA}(\boldsymbol{w}^{\text{test}})$ using the following equation:

$$v_{iso}(\boldsymbol{w}^{\text{test}}) = \frac{1}{K}\sum_{i=1}^{K} \Pi_{\text{Isomap}}(\boldsymbol{v}_{LDA}(\boldsymbol{w}_i^{\text{ref}})) \quad (2)$$

Using this algorithm, we can plot two curricula in the same map to satisfy Criterion 3.

3 Experiment

3.1 Curriculum Analysis based on CS2008

The Review Task Force commissioned by the ACM Education Board and the IEEE Computer Society's Education is now finalizing CS2001 as a reference curriculum

Table 2: CS2008 Knowledge Areas and Topics

1	**SE** software, system, engineering, development, tool, requirement, applicable, design, method, approach
2	**GV and HC** computer, graphic, image, information, technique, model, visualization, user, design, visual
3	**DS** structure, discrete, computer, science, theory, material, formal, proof, graph, topic
4	**AL and IM** algorithm, system, software, efficiency, programming, language, information, explain, performance, particular
5	**NC, IM, and HC** network, concept, computing, datum, protocol, system, application, web, technology, involve
6	**IS** system, learning, agent, planning, search, algorithm, method, reasoning, machine, AI
7	**SP** course, ethical, issue, computing, technical, student, social, context, understand, impact
8	**PF and PL** language, programming, computer, program, core, unit, paradigm, datum, science, basic
9	**AR and CN** computer, architecture, program, computing, topic, system, understand, level, student, performance
10	**OS and HC** system, operating, design, topic, implementation, internal, explain, hardware, device, algorithm

Table 3: MIT courses

1	6.930:Management in Engineering, 6.938:Engineering Risk-Benefit Analysis, 6.163:Strobe Project Laboratory
2	6.837:Computer Graphics, 18.965:Geometry of Manifolds, 6.801:Machine Vision
3	18.175:Theory of Probability, 18.103:Fourier Analysis - Theory and Applications, 18.315:Combinatorial Theory
4	18.415J:Advanced Algorithms, 6.854J:Advanced Algorithms, 6.830:Database Systems
5	6.263J:Data Communication Networks, 6.452:Principles of Wireless Communications, 6.829:Computer Networks
6	6.541J:Speech Communication, 6.034:Artificial Intelligence, 6.864:Advanced Natural Language Processing
7	6.901:Inventions and Patents, 6.071J:Introduction to Electronics, Signals, and Measurement, 6.912:Introduction to Copyright Law
8	18.725:Algebraic Geometry, 6.252J:Nonlinear Programming, 18.906:Algebraic Topology II
9	6.443J:Quantum Information Science, 6.453:Quantum Optical Communication, 18.435J:Quantum Computation
10	6.828:Operating System Engineering, 18.125:Measure and Integration, 6.823:Computer System Architecture

in computer science. The Task Force identified a set of 14 knowledge areas, each of which contains about 10 knowledge units which correspond to syllabi. Table 1 lists all knowledge areas.

We used LDA-C[5] which was a C-implementation of Blei's LDA model. With LDA-C we can estimate β from a set of documents and use it for later analysis as explained at Step 3 in the previous algorithm.

The top 10 words with the highest $p(w_n|z,\beta)$ for each topic and knowledge area names (in bold) are shown in Table 2. For example, Topic 1 corresponds to "SE:Software Engineering." The units of "HC:Human-Computer Interaction" are divided to Topic 2 and Topic 5, according to their relation to GUI, usability, collaboration, and so on. Topic 8 includes "PF:Programming Fundamentals" and "PL:Programming Languages" because the description of PF and PL is relatively short, and looks similar.

3.2 Analysis of Syllabi of MIT

We used the computer science-related course syllabi of Massachusetts Institute of Technology (MIT) as a first target of analysis. This is because MIT provides many computer science-related courses and its course syllabi are available at OCW[2]. As computer science-related courses, we picked up 299 courses of the departments of "Electrical Engineering and Computer Science" and "Mathematics" from 1999 to 2008. We extracted a text from "Course Home" and "Syllabus" web pages of each course, and eliminated obviously unnecessary words such as HTML tags, header and footer, stop words, and some frequent words specific to OCW.

All courses of MIT are related to several topics of CS2008. We used the probability $p(\boldsymbol{w}|z)$ as the degree how the syllabus $\boldsymbol{w}$ was related to the topic z. The top 3 courses with respect to $p(\boldsymbol{w}|z)$ are shown in Table 3. Figure 3 shows the all course syllabi, each of which is represented as a small circle and a course id in a different color according to the most strongly related topic. To locate topics in the figure, we used imaginary courses $\boldsymbol{w}$ with $p(\boldsymbol{w}|z) = 1$ for each topic z. Our method satisfies Criteria 1 and 3 because we can extract topics from CS2008 and map course syllabi of the MIT curriculum based on CS2008 topics.

From Table 3 and Figure 3, we can see the following characteristics of MIT curriculum. Relatively many courses are provided for Topics 3, 4, 5, and 9 (DS, AL, IM, NC, AR and CN) in MIT curriculum. This shows the emphasis of MIT curriculum. On the other hand, a few MIT courses are related to Topic 10 (OS).

Courses lie between the topics to which they are related and the origin, and courses which have a strong relation to a specific topic are located near to the topic. For example, "6.252J Nonlinear Programming" which is categorized in Topic 8 lies near Topic 3 courses. According to

[2] http://ocw.mit.edu/

Table 4: The OU courses

1	M885:Analysis and design of enterprise systems: an object-oriented approach, M865:Project management, T837:Systems engineering
2	TT280:Web applications: design, development and management, M150:Data, computing and information, M255:Object-oriented programming with Java
3	M263:Building blocks of software, M359:Relational databases: theory and practice, M366:Natural and artificial intelligence
4	M257:Putting Java to work, TT380:Databases within website design, M876:Relational database systems
5	T822:Multi-service networks: structures, T823:Multi-service networks: controls, TT382:Web server management, performance and tuning
6	M366:Natural and artificial intelligence, T226:ICTs, change and projects at work, T224:Computers and processors
7	T226:ICTs, change and projects at work, M226:Computing: a work-based approach, T161:Return to science, engineering and technology
8	M263:Building blocks of software, MT264:Designing applications with Visual Basic, M255:Object-oriented programming with Java
9	T224:Computers and processors, M366:Natural and artificial intelligence, T216:Cisco networking (CCNA)
10	TT281:The client side of application development, M873:User interface design and evaluation, M362:Developing concurrent distributed systems

the syllabus, this course provides a unified and computational approach to nonlinear optimization problems, which means that the course are related to not only theoretical issues but also practical issues. Because we can read this from the figure, our method satisfies Criterion 2.

3.3 Analysis of Syllabi of the Open University

Next, we compared the curriculum of MIT with the computer science-related course syllabi of the Open University (OU)[3]. The OU is the United Kingdom's only university dedicated to distance learning, and it offers over 600 courses which are categorized into 14 fields. We picked up 55 courses in "Computing and ICT courses". We extracted a text from "Summary " and "Course content" web pages of each course.

The top 3 courses with respect to $p(w|z)$ are shown in Table 4. Using the algorithm mentioned in Section 2.3, we plotted the course syllabi of the OU into the map of MIT generated in Section 3.2.

We first calculated the distance to the closest three other courses in a higher dimensional LDA topic space as shown in Figure 1. As for MIT course syllabi (shown in red "+") the distance between courses peaks at 18. As for the OU (shown in green "x"), the peak is 28. As to the closest three MIT courses to each OU course, the distance peaks at 30 (shown in "*"). Because there is no much difference between these distributions, we can expect the OU courses and properly mapped by the Isomap for the MIT courses.

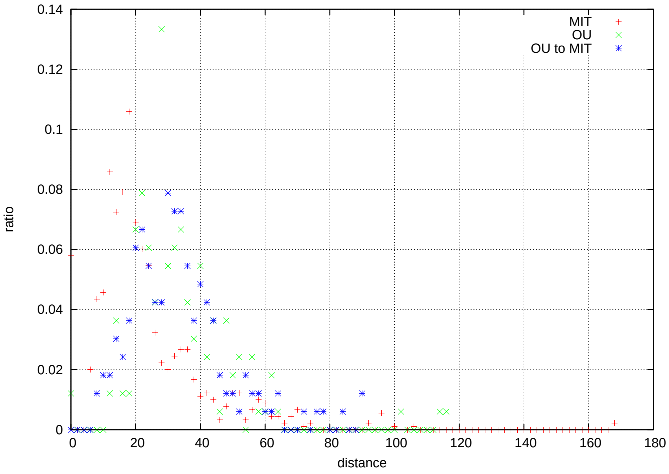

Figure 1: Distance to the closest courses

Figure 2 shows how the OU course (TT281) was mapped in the map of MIT. In a higher dimensional LDA topic space, three MIT courses (6.152J, 6.022J, 6.892) are closest to TT281. These three MIT courses are mapped to nearby points by Isomap, so the OU course is mapped to a point close to these nearby points.

Figure 4 shows all the courses of MIT (smaller circles) and OU (larger circles).

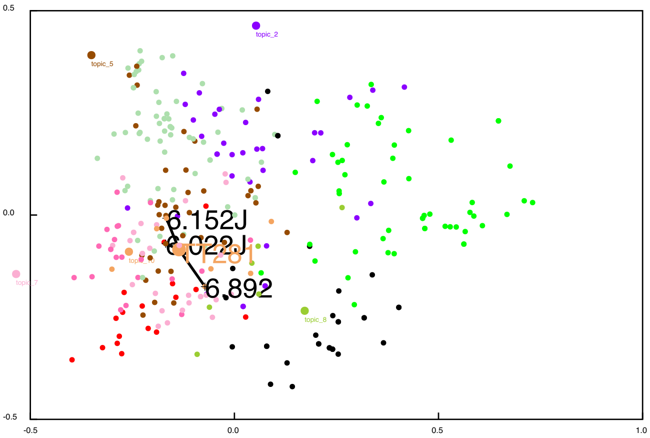

Figure 2: The plot of the OU course (TT281) in the MIT map

From these table and figure, we can see the characteris-

[3] http://www.open.ac.uk/

tics of the OU curriculum. Most OU course syllabi are plotted in the left area in the map of MIT. This means that many OU courses are provided for Topics 1, 5, and 7 (SE, NC, IM, HC, SP), and few courses are for Topic 3 (DS). This practical nature of courses in OU is consistent with the fact that about 70 percent of undergraduate students are in full-time employment in OU.

Since we generate a map of two curricula in the same space, we can find similar course syllabi from the map. For example, "T320: E-business technologies: foundations and practice" and "TT281: The client side of application development" in the OU curriculum are the courses on web technology. We can find the similar courses such as "6.171: Software Engineering for Web Applications" and "6.897: Selected Topics in Cryptography" in the MIT curriculum.

4 Related Work

Mima developed the MIMA search that uses automatic recognition techniques on technical words and clustering words[6, 7]. It generates a graph with words and syllabi as nodes and word-syllabus matrix as arcs. It is useful for browsing local relationships between words and syllabi; however, it lacks the global structure visualization function that our approach does. Ida et al. developed a course classification system using syllabus data[8, 9]. With their tool, users can analyze curricula interactively from various viewpoints. However, it does not automatically give a holistic view of the entire curriculum.

Tungare et al. created a repository system for computer science syllabi[10, 11]. They developed tools such as Syllabus- Maker for creating and comparing syllabi. They have not developed a technique for systematizing syllabi.

5 Summary

In this paper, we proposed using a method involving LDA to analyze syllabi and to construct a two-dimensional map from Isomap so that syllabi could be holistically understood. We applied our methods to CS2008 and syllabi of MIT and the OU. We found that characteristics of curriculum of these two universities could be detected using LDA model of CS2008, and that comparison between two curricula could be conducted through the unified map from Isomap. Our technique satisfied the criteria we set first.

Now, we are planning to extend our experiments to syllabi from other universities. We are expecting to find similarities and differences among curricula of different universities, and sparse fields which are not sufficiently educated.

References

[1] T. Sekiya and K. Yamaguchi. Ontology and context based repository of learning materials, 2007. Poster Session, CAL'07 DEVELOPMENT, DISRUPTION & DEBATE - D^3, Dublin, Ireland.

[2] Joshua B. Tenenbaum, Vin de Silva, and John C. Langford. A global geometric framework for nonlinear dimensionality reduction. *Science*, 290(5500):2319–2323, 2000.

[3] The Joint IEEE Computer Society/ACM Task Force. Computing curricula, 2008. http://wiki.acm.org/cs2001/.

[4] David M. Blei, Andrew Y. Ng, and Michael I. Jordan. Latent dirichlet allocation. *Journal of Machine Learning Research*, 3:993–1022, 2003.

[5] David M. Blei. Lda-c, 2006. http://www.cs.princeton.edu/˜blei/lda-c/.

[6] Hideki Mima. Mima search: a structuring knowledge system towards innovation for engineering education. In *Proceedings of the COLING/ACL on Interactive presentation sessions*, pages 21–24, Morristown, NJ, USA, 2006. Association for Computational Linguistics.

[7] Center for Innovation in Engineering Education, the University of Tokyo. Mima search, 2007. http://ciee.t.u-tokyo.ac.jp/MimaSearch/manual/.

[8] K. Miyazaki, M. Ida, F. Yoshikane, T. Nozawa, and H. Kita. On development of a course classification system using syllabus data. *Computational Engineering I (The symposium book of selected papers at ICOME 2003)*, pages 311–318, 2004.

[9] M. Ida, T. Nozawa, F. Yoshikane, K. Miyazaki, and H. Kita. Syllabus database and web service on higher education. In *7th International Conference on Advanced Communication Technology (ICACT2005)*, volume 1, pages 415 – 418, 2005.

[10] Manas Tungare, Xiaoyan Yu, William Cameron, GuoFang Teng, Manuel A. Perez-Quinones, Lillian Cassel, Weiguo Fan, and Edward A. Fox. Towards a syllabus repository for computer science courses. In *Proceedings of the 38th SIGCSE technical symposium on Computer science education*, pages 55 – 59, 2007.

[11] Xiaoyan Yo, Manas Tungare, Weiguo Fan, Manuel A. Perez-Quinones, Edward A. Fox, William Cameron, GuoFang Teng, and Lillian Cassel. Automatic syllabus classification. In *Proceedings of JCDL'07*, pages 440 – 441, 2007.

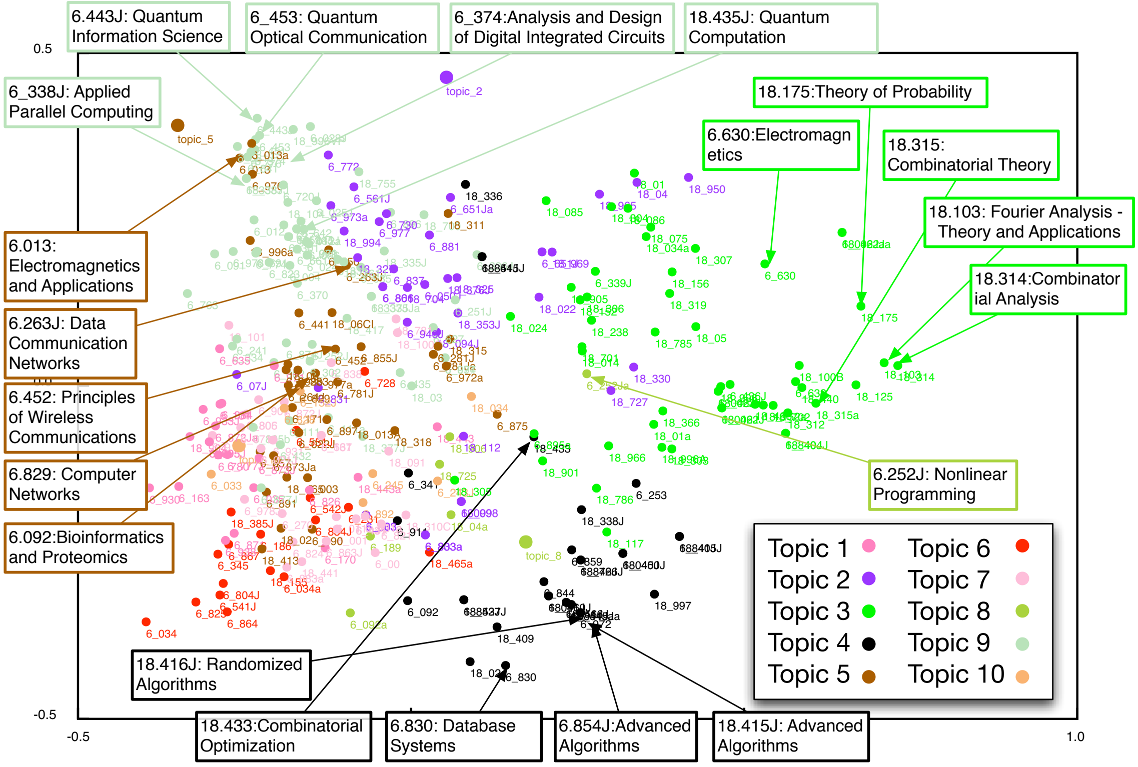

Figure 3: MIT courses

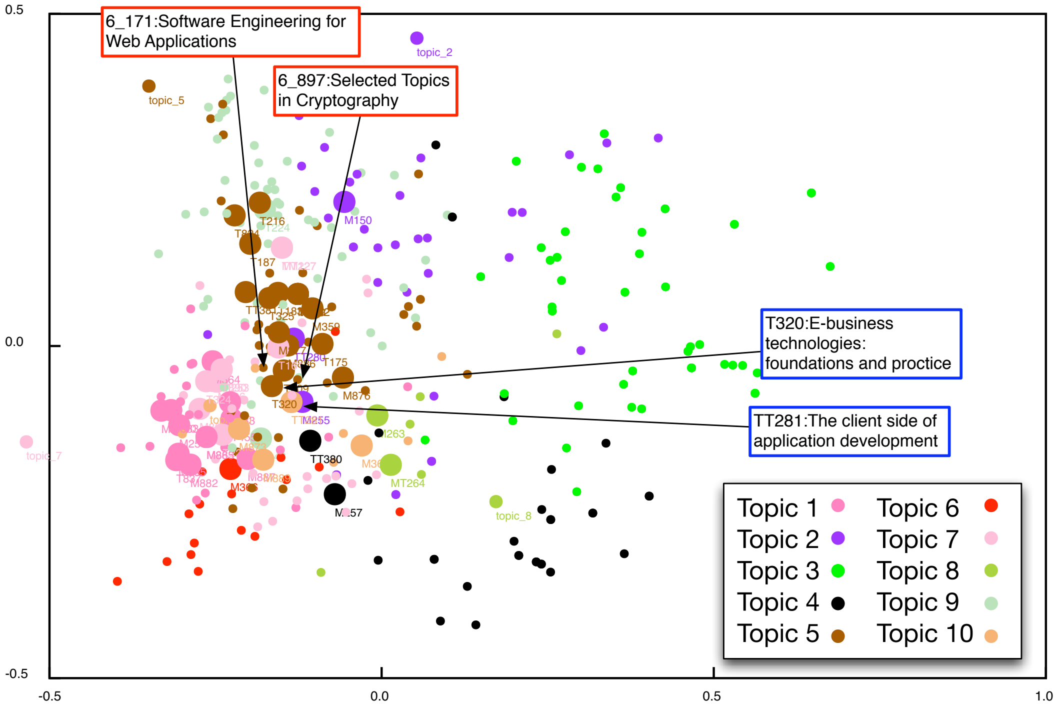

Figure 4: OU courses over MIT courses