# WHAT ARE WE TEACHING?

# Automated Evaluation of CS Curricula Content Using Topic Modeling

Jean Michel Rouly April 30, 2015

.

George Mason University

# **OVERVIEW**

- From 2007, number of new CS undergrads has increased 28.9%
- · Fifth straight year of increase

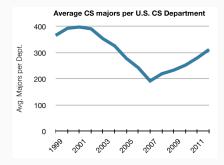


Figure 1: From Zweben, 2011.

# Relative Standing comparing against other, similar departments

Relative Standing comparing against other, similar departments Absolute Performance benchmark against standardized expectations (published by the ACM)

Relative Standing comparing against other, similar departments Absolute Performance benchmark against standardized expectations (published by the ACM)

Both of these require data about the topics covered in a course.

1. Widely available university course description data.

- 1. Widely available university course description data.
- 2. Descriptions detail what concepts are taught in a course.

- 1. Widely available university course description data.
- 2. Descriptions detail what concepts are taught in a course.
- 3. Human-readable descriptions require manual inspection.

- 1. Widely available university course description data.
- 2. Descriptions detail what concepts are taught in a course.
- 3. Human-readable descriptions require manual inspection.

So what to do?

Through the application of probabilistic machine learning methods, specifically LDA topic modeling, a corpus of unstructured course descriptions can be digested and mined for topics. In this scenario, each topic represents a core concept covered by the courses.

A research framework will be constructed to read data from the Internet, digest into a common internal format, pipeline into an LDA topic model, and ultimately visualize in an interactive manner.

Ultimately the automatically discovered topics can be used in end-user university evaluation processes.

Build a tool that automatically...

· ingests large collection of course descriptions

- $\cdot$  ingests large collection of course descriptions
- $\cdot\,$  infers topics from course descriptions

- $\cdot$  ingests large collection of course descriptions
- $\cdot\,$  infers topics from course descriptions
- · computes comparisons between departments

- $\cdot$  ingests large collection of course descriptions
- · infers topics from course descriptions
- $\cdot\,$  computes comparisons between departments
- $\cdot$  evaluates departments on their concepts

# BACKGROUND

# **Topic Modeling**

Attempts to discover the abstract topics of a dataset.

### Topics

A topic is a frequency distribution over terms, roughly representing a concept taught in a course.

#### Overview

Latent Dirichlet Allocation (LDA) is a form of *Latent Variable Modeling* that can infer topics from within a document.

LDA takes a generative approach to latent variable modeling, assuming the topics occur in some proportion within each document.

# DATA

University	Course Count	Web
American University	32	american.edu
George Mason University	145	gmu.edu
Kansas State University	83	ksu.edu
Louisiana State University	59	lsu.edu
Portland State University	190	pdx.edu
Rensselaer Polytechnic Institute	61	rpi.edu
University of South Carolina	64	sc.edu
Stanford University	69	stanford.edu
University of Utah	142	utah.edu
University of Tennessee, Knoxville	29	utk.edu
ACM Exemplar Courses	68	_

Table 1: University course descriptions

#### Raw Course Description

Capstone course focusing on design and successful implementation of major software project, encompassing broad spectrum of knowledge and skills, developed by team of students. Requires final exhibition to faculty-industry panel.

#### Cleaned course description

capston focus design success implement major softwar project encompass broad spectrum knowledg skill develop team student requir final exhibit faculti industri panel

# TRAJECTORY

**Trajectory** is a tool that automatically ingests course description data from the Internet and presents an accessible interface for departmental evaluation.

Lines of Python	3193
Lines of Java	631
Lines of HTML/CSS/JS	1828
Lines of JSON	3219
Lines of Bash	165
Size on disk	6.7M

Table 2: Code statistics

Four primary modules:

Scrape web-scrape online university catalogs Import/Export pass structured data between Learn and Scrape Learn estimate LDA topic model on data Web visualization tool

Underneath the entire system is a structured relational database layer.

Browsecollected data by university or departmentUnderstandcourses through inferred topicsAnalyzeconceptual overlap in prerequisite chainsComparedepartments based on conceptual compositionEvaluatedepartments against ACM benchmarks

# (Not So) Live Demo

Images have been modified to fit in this presentation.

Visit trajectory.rouly.net on your smart device to follow along.

# BROWSE

#	Department	Course Count	Web
1	Applied Information Technology	39	-
2	Computer Science	106	-

Figure 2: Browse a university's departments.

#### **Course List**

#	Number	Title
1	100	Principles of Computing
2	101	Preview of Computer Science
3	105	Computer Ethics and Society
4	112	Introduction to Computer Programming
5	123	Computing: From the Abacus to the Web
6	211	Object-Oriented Programming
7	222	Computer Programming for Engineers

Figure 3: Browse a department's courses.

# UNDERSTAND

# **GMU CS 105**

#### **Computer Ethics and Society**

/ UNIVERSITIES / GMU / CS / 105

Inferred Topics

• (0.759646) ethic, comput, issu, profession, social, technolog, privaci, legal, relat, digit



Figure 4: View a course's inferred topics.

#### Inferred Topics

(0.734343) comput, ethic, issu, social, profession, technolog, impact, privaci, properti, legal 34
(0.912) PDX CS 305 Social, Ethical, and Legal Implications of Computing
(0.833) SC CSCE 390 Professional Issues in Computer Science and Engineering
(0.734) GMU CS 105 Computer Ethics and Society
(0.680) KSU CIS 415 Ethics and Computing Technology
(0.670) LSU CSC 1200 Ethics in Computing
(0.629) Stanford CS 181 Computers, Ethics, and Public Policy
(0.625) SC CSCE 190 Computing in the Modern World

Figure 5: View related courses.

### Inferred Topics

- (0.748265) data, mine, algorithm, larg, web, effici, techniqu, search, time, cluster
- (0.748) RPI CSCI 6390 Database Mining
- (0.714) Utah CS 6140 Data Mining
- (0.695) RPI CSCI 4390 Database Mining
- (0.644) Utah CS 5140 Data Mining
- (0.619) Stanford CS 276 Information Retrieval and Web Search (LINGUIST 286)
- (0.614) LSU CSC 7442 Data Mining and Knowledge Discovery
- (0.588) GMU CS 674 Data Mining on Multimedia Data

### Figure 6: View related courses.

16

### Inferred Topics

• (0.402983) game, video, player, film, solut, graphic, emphasi, creat, anim, varieti

(0.703) PDX CS 442 Advanced Artificial Intelligence: Combinatorial Games

(0.660) PDX CS 542 Advanced Artificial Intelligence: Combinatorial Games

(0.509) Utah CS 6665 Character Animation

(0.433) Utah CS 6660 Physics-based Animation

(0.403) GMU CS 225 Culture and Theory of Games

(0.365) KSU CIS 580 Fundamentals of Game Programming

(0.351) RPI CSCI 4540 Game Development II

Figure 7: View related courses.

## ANALYZE

## TRAJECTORY / WEB / ANALYZE

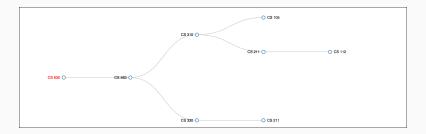


Figure 8: Course prerequisite tree.

## COMPARE

## TRAJECTORY / WEB / COMPARE

#### TOPICS IN GMU CS (9)

search, plan, infer, problem, inform, method, constraint, logic, state, satisfact (link)

student, assign, respons, question, conduct, cloud, paper, studi, answer, univers (link)

program, type, languag, compil, time, run, dynam, implement, static, machin (link)

theori, matter, question, subject, serv, continu, aris, commit, purpos, sophist (link)

compon, code, malici, weak, develop, softwar, java, student, common, opportun (link)

estim, linear, featur, reduct, nonlinear, filter, dimension, nonparametr, parametr, select (link)

activ, guidanc, singl, simultan, curricula, elect, action, consist, recipi, agreement (link)

multicor, perform, project, program, master, suitabl, gpu, purpos, achiev, potenti (link)

#### COMMON TOPICS (41)

comput, ethic, issu, social, profession, technolog, impact, privaci, properti, legal (link)

softwar, design, test, system, develop, requir, manag, model, engin, specif (link)

learn, decis, network, model, algorithm, topic, method, probabilist, neural, approach (link)

model, system, problem, perform, studi, analysi, evalu, simul, comput, implement (link)

speaker, seri, number, faculti, space, form, univers, field, prime, existenti (link)

student, class, week, modul, assign, hand, lectur, unit, lab, expect (link)

program, languag, object, orient, includ, type, design, abstract, data, specif (link)

data, mine, algorithm, larg, web, effici, techniqu, search, time, cluster (link)

class, test, program, condit, code, review, api,

#### TOPICS IN STANFORD CS (16)

interfac, design, user, interact, human, evalu, prototyp, graphic, hci, cognit (link)

case, studi, class, includ, learn, written, exam, assign, describ, code (link)

inform, compani, corpor, process, partner, worth, assur, risk, invest, point (link)

languag, natur, understand, process, model, analysi, method, answer, translat, question (link)

comput, investig, technolog, incid, role, base, understand, theori, practic, guid (link)

file, script, includ, linux, unix, exam, configur, manag, administr, mac (link)

includ, estim, theorem, probabl, distribut, condit, variabl, count, expect, independ (link)

multimedia, network, data, video, media, stream, prerequisit, audio, peer, sensor (link)

data, problem, state, big, set, present, law, polici,

Figure 9: Compare departmental concept coverage.

## TRAJECTORY / WEB / COMPARE

STATISTICS					
GMU Computer Science vs. Stanford Computer Science					
Similarity					
Euclidean Distance: 5.000					
Cosine Similarity [-1, 1]: 0.232					
Jaccard Index [0, 1]: 0.621					
Sizes					
Courses in GMU Computer Science: 106					
Courses in Stanford Computer Science: 69					
Topic Vectors					
Department A:					
1011110111000000110101101101101100000110000					
Department B:					
1001101100000010001010110111101000011010					

Figure 10: Quick departmental comparison.

## **EVALUATE**

## TRAJECTORY / WEB / EVALUATE

#	Number	Title	Knowledge Areas	Ground Truth
1	100	Principles of Computing	AL, CN, SP, IM	None.
2	101	Preview of Computer Science	CN	None.
3	105	Computer Ethics and Society	SE, SP	IAS, NC, SP
4	112	Introduction to Computer Programming	AL, CN, SP, IM	AL, AR, CN, PL, SDF
5	123	Computing: From the Abacus to the Web	SP	None.
6	211	Object-Oriented Programming	AL, CN, IM	AL, AR, CN, OS, PL
7	222	Computer Programming for Engineers	AL, CN, IM	PL
8	225	Culture and Theory of Games	SP	None.

Figure 11: Evaluate department against Knowledge Areas.

WRAP UP

- · Learning Outcomes meta-analysis
- $\cdot$  Alternative methods
- · Topic summarization
- $\cdot\,$  Extension to workforce preparedness
- · Smart web scraping

Try out **Trajectory** online at

### trajectory.rouly.net

### Get the source of this presentation and the Trajectory project at

## github.com/jrouly/trajectory

Trajectory is licensed under the Apache version 2.0 license.

Co-authors:

- · Huzefa Rangwala
- · Aditya Johri

Presentation theme:

github.com/matze/mtheme

- D. M. Blei, A. Y. Ng, and M. I. Jordan. Latent Dirichlet Allocation. J. Mach. Learn. Res., 3:993–1022, Mar. 2003.
- ACM/IEEE-CS Joint Task Force on Computing Curricula. Computer science curricula 2013. Technical report, ACM Press and IEEE Computer Society Press, December 2013.
- S. Zweben. Computing degree and enrollment trends. Technical report, Computing Research Association, 2011.

# QUESTIONS?