History 100S, Spring 2017

Text Analysis for Digital Humanists and Social Scientists

Instructor

Laura K. Nelson
Email: Lknelson3@berkeley.edu
Twitter: @LauraK_Nelson
Office hours: I will hold office hours in the BIDS space in 190 Doe Library. You will be required to sign-in when you enter the space.

Mondays, 1:30PM-2:30PM (general questions), **190D** Doe Library
Wednesdays, 1:30PM-2:30PM (programming questions), **190C** Doe Library

Course Description

Meetings: Mondays and Wednesdays, 10AM-12PM, 458 Evans

Readings:

All of the readings are available online. I provide the link to these readings in the syllabus, which are also available on the bCourses site. If you are having problems accessing these readings let me know as soon as possible.

Overview:

Increasingly, humanity's cultural material is being captured and stored in the form of electronic text. From historical documents, literature and poems, diaries, political speeches, and government documents, to emails, text messages, and social media, students from the humanities and social sciences now have access to immense amounts of rich, and diverse, text. Scholars are increasingly using computational methods to analyze these new sources of text in order to ask, and answer, a diverse array of questions about the social world: Does social media reflect public political opinion, or drive it? What determines trust in online communities? What types of blog posts get censored in China and why? Are diurnal and seasonal mood cycles cross-cultural? What was the form of cultural and institutional change through the "civilizing process" in England between the 16th and 20th centuries? What is the life cycle of a literary genre? What are textual allusions in Classical Latin poetry? Can the FBI *really* analyze 650,000 emails in 3 days? (Spoiler: Yes, they can!)

In this course you will learn cutting-edge methods to analyze large amounts of texts to explore questions fundamental to the humanities and social sciences. We will not have computers read the text for us. Instead, we will harness the superior ability for computers to count and extract patterns from text, and use this output to enhance our own critical thinking and interpretive analyses. To implement these methods we will use the open source (and free!) programming language Python and the Jupyter platform. Specific skills covered include structuring and pre-processing text, dictionary methods, supervised and unsupervised machine learning, word scores and word weighting, grammar-parsing and concordances, working with metadata, and crowd-based content analysis. The ultimate goal is to encourage you to think about novel ways you can apply these techniques to your own text and research questions. By the end of the course you will have a better understanding of the range of text analysis techniques available, what kind of evidence the different techniques produce, how this evidence can be used to help you better understand the social world, and how to use this evidence to persuade others of your interpretation.

In this course you will implement text analysis techniques in Python to carry out an applied, real-word research project. You (on your own or in a team) will process, analyze, and interpret a corpus of your choosing to answer a question about the social world that has not yet be explored. I will provide demonstration corpora relevant to both the humanities and social sciences, but I encourage students to bring their own corpus if desired. The corpora are chosen to allow for a variety of questions and disciplinary frameworks. The form of the final project will be a Jupyter notebook. A Jupyter notebook is an interactive computational environment that allows you to combine text, code, output, and visualizations into one document, and easily share the document with colleagues or publish it on the web. It can be used with a variety of programming languages, including Python. Because it is a functioning program environment which also can incorporate text and visualizations in a seamless and visually pleasing manner, it is popularly used to teach programming and computational methods, to present scientific findings, and in industry, including data-driven journalism. Here's an example of a Jupyter notebook, and a good one to emulate for your own project:

http://nbviewer.jupyter.org/github/brianckeegan/Bechdel/blob/master/Bechdel_test.ipynb

The skills required to ask and answer a question about the social world using computational methods and present the findings in a cohesive document are widely used in both academia and industry. This course will facilitate career paths in each.

This class is designed for two types of students: advanced undergraduates, graduate students, and faculty from 1) the humanities and social sciences with no or little prior knowledge of text analysis and computational methods, and 2) those from computer science or statistics with no or little prior background in the humanities and social sciences. No computer programming knowledge is assumed, but this course is also appropriate for those who already have a working knowledge of Python or other programming languages and/or have foundational knowledge in applied math, algorithms, and statistics (for example those who have taken CS61A, CS10, or Data8 courses), and are looking for ways to apply their knowledge to specific domains outside of the physical sciences. This course will also build on courses like History 104: The Craft of History.

The course will meet for 4 hours of class time each week. Class time will be a combination of lectures, discussion, and hands-on tutorials. Students’ grades will be based on class participation, reading responses, programming assignments, a project proposal and presentation, and a final project and presentation.

Course Requirements

Technology Requirements

Students must have access to a laptop, and you must bring it to class every day. If you do not have a laptop contact me and we can try to work something out.

This workshop will be taught in the open source programming language Python and the programming environment Jupyter. **Participants should install** Anaconda for Python 3.5 **on their laptops prior to Monday, January 23** (https://www.continuum.io/downloads). Anaconda includes Python, the necessary Python packages, and Jupyter.

Grading and Assignments

10% Attendance and Participation:
- 5% - Attendance + weekly survey
- 5% - Active participation in class discussion and tutorials

15% Five (5) short reading responses, posted to the class website by 7PM the night before class. Students may choose which days to submit a response, but at least two must be during the first four weeks of the course.

15% Two (2) programming assignments, #1 in class February 22, #2 due by 7PM on April 9.

20% Two in-class peer-reviewed check-ins, on February 22 and April 24.

40% Final Project and Presentation:
- 10% - Research Proposal, due by 7PM on March 14 and presented on March 15
- 30% - Final Project, due by 3PM on May 9
  In-person presentations 3PM-6PM, May 9

Course Schedule

Session	Date	Theme	Notes
Week 0	January 16 January 18	**Academic Holiday, No Class** Introduction and overview	guest lecture
Week 1	January 23 January 25	“There Will Be Numbers”	**install Anaconda on your laptop prior to Monday's class**
Week 2	January 30 February 1	Texts and numbers	
Week 3	February 6 February 8	Words, syntax, and part of speech	
Week 4	February 13 February 15	Working with text as data Distinguishing texts by word use	
Week 5	February 20 February 22	**Academic Holiday, No Class** **Check-In #1: Research Questions**	**Bring Assignment #1 to class**
Week 6	February 27 March 1	Measuring known categories	
Week 7	March 6 March 8	Digitizing your own corpus Preparing your corpus for analysis	
Week 8	March 13 March 15	Theoretically-informed research **Project Proposal Lightening Talks**	in-class presentations
Week 9	March 20 March 22	Scaling up from labeled texts	
Week 10	March 27 March 29	**Spring Break**	no class
Week 11	April 3 April 5	Uncovering patterns in text	
Week 12	April 10 April 12	Words and their context Visualizing text analysis	**assignment #2 due by 7PM, 04.09**
Week 13	April 17 April 19	Computational hermeneutics The power of the crowd	guest lecture
Week 14	April 24 April 26	**Check-In #2: Preliminary Analysis** Wrap-up discussion	**bring completed notebook**
Week 15	May 1 May 3	R/R/R Week	I will hold office hours both days.
Final Exam	May 9	**Final Project Lightening Talks**	**Confirm exact date and time prior to the week.**

Course Structure

This is a hands-on course that will introduce students to the practical application of computational text analysis methods to questions important to humanists and social scientists. In this course you will learn what types of questions humanists and social scientists are answering using text analysis techniques, what types of evidence these techniques produce, how you can use this evidence to draw conclusions about the social world, and you will learn the programming language Python and the developing environment Jupyter through the applied use of text analysis techniques. To facilitate learning both programming and domain knowledge, the course will consist of practical tutorials aimed at getting you processing and analyzing text via Python, as well as discussions about assigned readings that explore a practical question using these techniques. It is important that you complete the readings before each class and come prepared to discuss the material. During these discussions there will be space to critique the material and these methods. It is important that we respect one another's thoughts, give everybody the space to talk, and address our comments at the ideas and not the person.

In each week we will learn skills and develop knowledge that build on previous skills learned, so it is important to attend every class. You are allowed one (1) absence before your grade is affected. If you know you are going to miss a week you should notify me at least four days in advance and I will let you know if the absence will be excused. If you miss a class you should make arrangements with a classmate to go over the material.

Assignments

I will not accept any late assignments. The Reading Responses are due by **7PM** the night before class. The first programming assignment should be brought to class on Wednesday, February 22. Assignment #2 is due by **7PM**, Sunday April 9. I will check the class website promptly at 7PM and I will not excuse late assignments because of technical difficulties of any kind, so plan ahead and post early. The two in-class check-ins are meant to help you build toward your final project, and will allow you to work through any challenges you have in a supportive environment. In lieu of a final exam, you will be required to ask and explore a question relevant to the humanities and/or social sciences using text as data and computational techniques. The final project is designed to encourage you to creatively combine the knowledge and skills built through the semester to explore a question about the social world that has not yet been answered, or to explore an old question in a new way. The form of the final project is a Jupyter notebook that contains Python code and output, as well as your interpretation of the output and conclusion about the real world (see here for an example of a Jupyter notebook). I will hand out a detailed rubric closer to the due date for each assignment and check-in, outlining exactly what is required. You may do your class project on your own or in a small team (maximum 3 people). If you choose to work in a team you will be required to submit a document with your project proposal detailing the planned division of labor and a shared contract, and another document with the final project revisiting these plans and collectively determining if they were met. You will present your Jupyter notebook to the class during the scheduled final: 3PM-6PM on May 9. For team projects, each team member must participate in the presentation.

Attendance and Participation

You are required to come to every class, as each class period builds on previous ones. You will be asked to fill out a survey at the end of class on Wednesdays. This survey will allow you to reflect on what you learned during the week, and will help me tailor the class to your needs. The survey will contribute to your attendance grade. You are also required to actively participate in discussions and in tutorials.

Reading Responses

In this course humanities and social science questions are central, and computational methods are used to answer those questions. There will be assigned readings that discuss either the theory behind these methods, or their practical application to real-world questions. You are required to submit five (5) short (two-paragraphs to one-page) reading responses, posted to the class website by 7PM on the night before class. You may choose which days to submit a response, but at least two (2) should be in the first four weeks of the course. In your reading response you should compare themes within the week's readings to each other and to previous readings in the course, and offer some reflections (critical or not) on the readings. I will provide reading questions to go along with each reading that you may use to guide your response. You may also relate the readings to the project you are developing, to help you prepare your final project. These short responses will help you understand and evaluate the applied use of these methods, they will help you get used to writing and talking about computational methods and the different types of evidence these methods produce, and they will help frame the class discussion. I encourage everyone to read each other's responses before class

You can get up to 2 points per reading response. One point for submitting a response, and up to one point for the thoughtfulness of the response.

Programming Assignments

You will also be required to complete two (2) programming assignments in the form of Jupyter notebooks, due in class on February 22 (week 5), and by 7PM on April 9 (week 12).

You can get up to 3 points per assignment. One point for submitting a notebook, one point if your code runs correctly, and up to one point for the substantive portion – how you interpret the output.

Mid-Term Check-Ins

There will be two in-class, peer-reviewed, mid-term check-ins, on February 22 (Week 5) and April 24 (week 14). The check-ins are meant to ensure that you are understanding the material (and that I am successfully teaching the material!), and to help you work toward your final project.

For the first check-in (February 22), you will be asked to bring a short programming assignment to class, in the form of a Jupyter notebook. You will then demonstrate your notebook to a peer, in turn, and then 1) walk each other through your thought process and why you did what you did to solve the problem, and 2) together come up with two substantive questions that might be addressed by the technique(s) mentioned in the assignment. By the end of the class you will be required to submit both of your programming solutions, a brief description of the differences in your two approaches and the benefits/drawbacks of each, and the two substantive questions you came up with.

The second check-in (April 24) will include two preliminary analyses for your final project. You should come to class with a Jupyter notebook that implements, or attempts to implement, two different analyses on your chosen corpus and a description of what you hope to accomplish with these analyses (if you chose to work in a team, each individual will have to write their own notebook for this check-in). You will then be paired with another classmate to review each other's notebooks and help each other with any challenges you face or areas you are stuck. You will then evaluate each other on the appropriateness of the technique to the chosen question. The goal here is not to grade your classmate (I will be giving the grade), but to practice evaluating the application of text analysis techniques to real-world questions in order to improve your own project and help your peer improve theirs. By the end of the class period you will be required to submit the evaluation of your classmate along with the two analyses you implemented (which, time permitting, you will be able to revise in class, based on your classmate's suggestions).

Project Proposal

The goal of the final project is to creatively combine the techniques you learned in the course to explore a question related to the humanities or social sciences that has either not been addressed before, or explore an old question in new ways. Through this project you should show that you understand (a) what types of questions are interesting or important to humanists and/or social scientists, (b) what types of questions can be best answered using computational text analysis methods, (c) what types of techniques and evidence are appropriate to best answer your question, and (d) that you can present your findings and analysis in a reproducible way and in a way that supports, and persuades others of, your conclusion.

The project proposal will be a Jupyter notebook detailing a preliminary plan for your final project. You should include the following in your project proposal:

1. Identify a general question related to the humanities or the social sciences that you plan to address in your final project. You should outline why this is an interesting or important question and describe why computational methods are necessary and/or helpful in exploring this question. If possible, explain how others have answered/attempted to answer this question using different methods.

2. Identify the corpus, or collection of texts, you will use to explore this question, and briefly describe why the identified corpus is appropriate. Additionally describe how you will collect the corpus and whether or not it will need to be cleaned prior to analysis.

3. Describe the techniques you expect to use to analyze the text and explore your question. Why these techniques and not others? What kind of evidence will these techniques

produce, and how will this help you answer the question and persuade others of your answer? If you already have some preliminary analyses, include these as well.

4. Briefly discuss any data visualization or interpretive techniques you will use to present your findings and convince others of your interpretation.

5. If your project will be a team project, detail a planned division of labor and a shared contract about your collective expectations (I encourage you to revisit this document well before the final project is due, to check-in with each about about whether the shared contract is working). The best teams will include people from different disciplinary backgrounds so you can leverage each other's specialized knowledge, e.g. a computer scientist and a historian.

Final Project

Keeping the above goals in mind, the final Jupyter notebook should include the following:

1. 1-2 cells describing the question or puzzle you are exploring, why it is interesting or important, how others have attempted to answer this question, and how you are improving on these answers. If no one has addressed this question, explain why you think this is the case. In other words, what are you doing that's different than what others have done?

2. 2-4 cells describing the corpus you are using to explore the question and how you collected the corpus. These cells should include summary statistics of the corpus, including number of documents, word counts, summary of metadata, or other relevant information. If appropriate, describe what your corpus is representative of.

3. 5-10 cells containing the analysis. These cells should contain a description of the analysis process and why it is appropriate for your question and text, followed by code implementing the techniques, output from the calculations, and a description of how you understand the output.

4. 1-2 cells producing some sort of data visualization or data summary output.

5. 1-2 cells detailing your interpretation of the data and output, and broader conclusions about history and the world around you that you draw from your exploration. Support your interpretation with evidence from your analysis. End with suggestions for further analyses and other texts that could help us continue to explore your question.

6. If you worked in a team, revisit the planned division of labor and your shared contract that you submitted in the project proposal, and briefly describe whether you think you collectively lived up to the contract and why. If you would like to send me private thoughts about your team you can do so as well.

Questions? Discussion Board, Office Hours, and Email

If you come across errors as you run code that you can not solve through Googling post them to the discussion board (start a new thread for new errors). You may also post questions or comments about the readings or about your final project. I encourage everyone to answer each other's questions, as this is the best way to learn complicated material. Often many people will get the same error or will have similar questions, so check the discussion board for answers before posting your error or question. This is not the comments section on YouTube, so keep your comments respectful. Disrespect will absolutely not be tolerated. You are also encouraged to come to my office hours. Mondays are devoted to substantive questions about the readings or your final project. Wednesdays are devoted to programming questions. **Email should only be used for quick logistical questions or if you need to inform me of a planned absence.** I will get back to emails within 16 working-hours, so plan ahead.

My general philosophy is to work hard during the week, and to take weekends off. If you email me or post questions on a Friday afternoon or a weekend, I may not respond until the following Monday. Plan accordingly, as the programming assignments are due Sunday nights.

Consulting Resources

There are a number of consulting services offered on campus to help students with programming questions and research design questions. I encourage you to take advantage of these resources and in doing so, meet others on campus doing this type of work. These institutes are good resources more generally for students interesting in exploring these topics further.

Digital Humanities @ Berkeley: http://dh.berkeley.edu/consulting
D-Lab: http://dlab.berkeley.edu/consulting
BIDS: https://bids.berkeley.edu/resources/office-hours

Student Learning Center (SLC)

The final project includes a writing component. The Student Learning Center (SLC) Writing Program works under the assumption that all writers, regardless of their experience and abilities, benefit from informed, individualized, and personal feedback on their writing. Tutors are trained to work with non-native speakers of English and with writers from a variety of disciplines.

For more information call 510-642-7332 or visit http://slc.berkeley.edu/writing.

Note on Plagiarism

I encourage you to work together to help each other review the readings and to learn the coding. However, *all written work must be your own*. I take academic honesty seriously, and you should too.

For more information on your rights and responsibilities as a student see: http://campuslife.berkeley.edu/conduct.

Readings and Schedule

Note: Subjects covering practical programming skills rendered in *italics*. You will need your laptop for these subjects. Otherwise, no open screens are allowed.

This syllabus is subject to change over the semester. Always check the bCourses site for the most up to date information and for announcements.

Week 0: Introduction and Overview

Monday 01.16: Academic holiday, no class.

Wednesday 01.18
- Introduction and Overview: Goals for the Semester

****Assignment for Monday, 01.23: Install Anaconda on your own laptop****

Week 1: "There Will Be Numbers" and Computational Basics

In the first week we have two goals: convincing ourselves about the benefits there might be to using computational methods to better understand the social world, and we will learn the basics of Python. For those new to the humanities and social sciences, this week will introduce you to some core concepts from these fields. For those new to programming, this week will get you up and running with some basic foundational programming knowledge.

Monday 01.23
- Course expectations
- Introduction to computational thinking
- What is a Jupyter Notebook?
- *Tech Overview: Using Python and Jupyter*
- *Installation check*
- *Primer in Markdown*

  Readings:
  - Andrew Piper (2016). "There Will be Numbers." *Journal of Cultural Analytics*.

Wednesday 01.25
- Text Analysis in the humanities and the social sciences
- How to read books and articles in the humanities and social sciences
- *Python Basics*

  Readings:
  - Lev Manovich (2016). The Science of Culture? Social Computing, Digital Humanities and Cultural Analytics. *Journal of Cultural Analytics*.
  - Manuel Vallée (2011). Critical Reading in the Social Sciences. University of California, Berkeley UC Regents.

Week 2: Working with Texts and Numbers in Python

This week we will get more comfortable working with text and numbers in Python. We will also get an overview of text analysis methods in the social sciences (Monday), and on how to read articles containing statistical models (Wednesday). There is no way to teach statistics in one day. Rather, the goal is to get you to a point where you understand the basic vocabulary, you can competently read articles using statistics, and can interpret the graphs and output from a variety of models. If you want to learn more about statistics, there are many wonderful courses you can take here at Berkeley.

Monday 01.30

- Overview of text analysis techniques
- *Working with texts in Python*

Readings:

- H. Andrew Schwartz and Lyle H. Ungar (2015). Data-Driven Content Analysis of Social Media: A Systematic Overview of Automated Methods. *Annals of the American Academy of Political and Social Science* 659: 78-94.

Wednesday 02.01

- Primer on statistics
- *Working with numbers in Python*

Readings:

- Jeremy Freeze, Brian Powell, and Lala Carr Steelman (1999). Rebel Without Cause or Effect: Birth Order and Social Attitudes. *American Sociological Review* 64: 207-231.
- Greta Krippner (2000). How to Read a (Quantitative) Journal Article. *American Sociological Association Introduction to Sociology Series*.

Week 3: Words, Syntax, and Grammar

Natural language processing is the bedrock of all text analysis techniques. In a general and non-technical sense, natural language processing involves techniques that incorporate morphology into the analysis, such as syntax and grammar. This week we will cover the basics of natural language processing and learn how to implement a few key techniques in Python, including pre-processing, word frequencies, concordances, part of speech tagging, and named entity recognition.

Monday 02.06

- What is Natural Language Processing?
- *Python's NLTK and Counting Words*

Readings:

- Elizabeth Kolbert. 2016. "Our Automated Future: How long will it be before you lose your job to a robot?" *New Yorker* December 19 & 26 issue. Read only the first section (up until the end of the sentence "I for one welcome our new computer overlords.")

- Mark Algee-Hewitt, Sarah Allison, Marissa Gemma, Ryan Heuser, Franco Moretti, and Hannah Walser (2016). Canon/Archive. Large-scale Dynamics in the Literary Field. *Pamphlets of the Stanford Literary Lab*.

Wednesday 02.08

- History and uses of natural language processing
- *POS Tagging and Named Entity Recognition using NLTK*

Readings:

- Matthew Wilkens (2013). The Geographic Imagination of Civil War Era American Fiction. *American Literary History* 25 (4): 803-840.

Week 4: Working with Text as Data

NLTK is one way to process text in Python, but it is often combined with other packages to do more powerful types of text analysis. This week we will cover two Python structures that, often in combination with NLTK, allow us to expand our text analysis repertoire, the Pandas dataframe and the scikit-learn Document Term Matrix. We will discuss the benefits and drawbacks of each.

Monday 02.13

- Using data in interpretive analyses
- *The Pandas Dataframe and Descriptive Statistics*

Readings:

- Franco Moretti (2013). Operationalizing. *New Left Review* 84.

Wednesday 02.15

- Comparing corpora by word use
- *Document Term Matrix and tf-idf scores using scikit-learn*
- *Finding distinctive words*

Readings:

- Sara Klingenstein, Tim Hitchcock, and Simon DeCeo. (2014). **The civilizing process in London's Old Baily**. *Proceedings of the National Academy of Sciences* 111(26): 9419-9424.

Week 5: Assignment and Check-In

Monday 02.20: Academic holiday, no class! Take a break. Or work on Assignment #1. Better to take a break. If you've read this far in the syllabus start a discussion thread on bCourses or Piazza, or join a discussion thread if started, to post a picture of your favorite baby animal.

Wednesday 02.22: Check-In #1

- ****Bring Assignment 1 to class****

Week 6: Measuring Known Categories – Dictionary Methods

Dictionary methods are a simple way to measure different types of language used in text. We'll cover standard dictionaries and derived dictionaries, and why you might choose one over the other.

Monday 02.27

- What is a dictionary and how is it used to measure themes and discourse?
- *Counting words*

Readings:

- J. Eric Oliver and Wendy M. Rahn (2016). Rise of Trumpenvolk: Populism in the 2016 Election. *The Annals of the American Academy of Political and Social Science*, 667: 189-206.
- **Optional**: Justine Kao and Dan Jurafsky (2012). A Computational Analysis of Style, Affect, and Imagery in Contemporary Poetry. *Workshop on Computational Linguistics for Literature*, 8-17.

Wednesday 03.01

- Deriving dictionaries from labeled text
- *Distinctive phrases in texts*

Readings:

- Jacob Jensen, Ethan Kaplan, Suresh Naidu, and Laurence Wilse-Samson (2012). Political Polarization and the Dynamics of Political Language: Evidence from 130 Years of Partisan Speech." *Brookings Papers on Economic Activity*.

Week 7: Collecting and Digitizing a Corpus

For the next two weeks we're going to take a step back from text analysis techniques and think about collecting texts, developing questions, and *operationalizing* variables, or defining concepts, from text.

This week, collecting a corpus. Before we can analyze text, we first must collect it, digitize it, and structure it. So far we've been using a corpus that has already been processed and digitized. This week you'll learn how to do this step yourself. First, we will learn about text encoding. Second, we will learn about regular expressions. You will find that you will use regular expressions on an almost daily basis as you work with text. We have already been using both encoding and regular expressions, but we will formally address these this week.

Monday 03.06

- Digitizing a corpus: Guest lecture by Adam Anderson, Digital Humanities @ Berkeley

Wednesday 03.08

- *Encoding, and dealing with encoding in Python*
- *Regular expressions, structuring unstructured texts, pre-processing*
- Small group discussions about project proposal progress

Readings:

- David C. Zentgraph (2015). What Every Programmer Absolutely, Positively Needs to Know About Encodings And Character Sets to Work With Text.

Week 8: Constructing a Theoretically-Informed Project

How do you construct a digital humanities or social science project that uses text as data? You need a question, you need a hypothesis or an expectation, you need to operationalize or define variables or concepts, and of course you need a corpus, covered last week. For Monday we will read a paper that combines many of the techniques already covered. This will test your ability to understand and evaluate papers that implement text analysis techniques on social, text-based data. We will also reserve time on Monday so you can work on your project proposal and to ask questions.

Monday 03.13

- Coming up with interesting, theoretically-informed questions
- Sections to include in a research paper
- *Sandbox: time to work on your own project*

Readings:

- Dan Jurafsky, Victor Chahuneau, Bryan R. Routledge, and Noah A. Smith (2016). Linguistic Markers of Status in Food Culture: Bourdieu's Distinction in a Menu Corpus. *Journal of Cultural Analytics*.

****Project Proposal Due by 7PM Tuesday, 03.14****
Wednesday 03.15: Project Proposal Lightening Talks

Week 9: Scaling Up from Labeled Texts

There has been a massive increase in the development and use of machine learning algorithms in the past five years, and these techniques are being incorporated into multiple scientific disciplines as well as industry. The next two weeks will explore the use of machine learning in text analysis.

Monday 03.20

- Introduction to Machine Learning

Readings:

- Alex Smola and S.V.N. Vishwanathan. (2008). “Chapter 1: Introduction.” Pp. 3-36 in *Introduction to Machine Learning*. Cambridge: Cambridge University Press.

Wednesday 03.22

- *Supervised Machine Learning using scikit-learn*

Readings:

- Gary King, Jennifer Pan, and Margaret E Roberts (2013). How Censorship in China Allows Government Criticism but Silences Collective Expression. *American Political Science Review* 2 (107): 1-18.

Week 10: Spring Break!
If you've read this far in the syllabus well done! Start a discussion thread in bCourses or Piazza, or join one that is already started, and post a picture of your ideal vacation location.

Week 11: Uncovering Patterns in Text

Monday 04.03

- Topic modeling and culture
- *Topic modeling using scikit-learn*

Readings:

- Paul DiMaggio, Manish Nag, and David Blei. (2013). Exploiting affinities between topic modeling and the sociological perspective on culture: Application to newspaper coverage of U.S. government arts funding. *Poetics* 41(6): 570-606.

Wednesday 04.05

- Using clustering tools from the 1990s to study history
- *Document similarity and clustering using scikit-learn*

Reading:

- Ben Schmidt (2011). Machine Learning on High Seas.

****Assignment 2 Due by 7PM, 4.09****

Week 12: Words and Their Context

The vector space model is a more recent development in text analysis. We'll explore this using word2vec, a technique that incorporates the context of a word into the analysis. In the second class we'll explore a few techniques to visualize data.

Monday 04.10

- Vector Space Models
- *word2vec*

Readings:

- Ben Schmidt (2015). Rejecting the Gender Binary.
- **Optional**: Ben Schmidt (2015). Word Embeddings.

Wednesday 04.12

- Persuasive Visualization
- *Visualizing data using matplotlib*

Readings:

- Explore Ben Schmidt's Maps & Visualizations Gallery.

Week 13: Research Design and The Power of the Crowd

This week we'll explore different takes on the future of text analysis, some critical. On Monday we discuss meta-theories on the role of text analysis in understanding culture. On Wednesday we'll hear about a few ways scholars are incorporating crowdsourcing into the text analysis workflow.

Monday 04.17

- What is the role of text analysis in understanding culture?

  Readings:
  - Tara McPherson (2015). Designing for Difference. *differences: A Journal of Feminist Cultural Studies* 25(1): 177-188.
  - John W Mohr, Robin Wagner-Pacifici, Rondald L Breiger (2015). Toward a computational hermeneutics. *Big Data and Society*.

Wednesday 04.19

- Using crowdsourcing in text analysis: Guest lecture by Nick Adams, BIDS

  Readings: TBA

Week 14: Research Design and Course Wrap-Up

Monday 04.24: Check-In #2 (bring a completed Jupyter Notebook to class)

Wednesday 04.26

- Course Wrap-Up
- What next?
- Check-in on final projects

Week 15: R/R/R Week

Optional office hours, Monday 05.01 and Wednesday 05.03, 1:30-2:30 PM

Final Project Due before 3PM, Tuesday May 9

Final Project Lightening Talks Tuesday, May 9, 3-6PM.