Natural Language Processing

Syllabus

DIGS 20006 / 30006	*Instructor*: Jeffrey Tharsen tharsen@uchicago.edu
MWF 9:30-10:20	*Office Hours*: Fridays noon-2pm, or by appt. Regenstein Library 216
Social Sciences Research Building 401	*Office Phone*: (773) 834-5534

Course Description

Natural Language Processing (NLP) is a rapidly developing field with broad applicability throughout the hard sciences, social sciences, and the humanities. The ability to harness, employ and analyze linguistic and textual data effectively is a highly desirable skill for academic work, in government, and throughout the private sector.

This course is intended as a theoretical and methodological introduction to a the most widely used and effective current techniques, strategies and toolkits for natural language processing, with a primary focus on those available in the Python programming language.

We will also consider how harnessing large digital corpora and large-scale textual data sources has changed how scholars engage with and evaluate digital archives and textual sources, and what opportunities textual repositories offer for computational approaches to the study of literature, history and a variety of other fields, including law, medicine, business and the social sciences.

In addition to evaluating new digital methodologies in the light of traditional approaches to philological analysis, students will gain extensive experience in using Python to conduct textual and linguistic analyses, and by the end of the course, will have developed their own individual projects, thereby gaining a practical understanding of natural language processing workflows along with specific tools and methods for evaluating the results achieved through NLP-based exploratory and analytical strategies.

Throughout this course, the sources, methodologies and tools we will focus on will be in part decided by student interests and goals, so as we progress, please take note of and send to me any specific types of toolkits or approaches you think might be useful or relevant for your work and analyses. Suggestions or ideas you have on approaches to NLP and other related topics we address in the course are welcome at any time.

Course Goals

Students who complete this course will gain a foundational understanding in natural language processing methods and strategies. They will also learn how to evaluate the strengths and weaknesses of various NLP technologies and frameworks as they gain practical experience in the NLP toolkits available. Students will also learn how to employ literary-historical NLP-based analytic techniques like stylometry, topic modeling, synsetting and named entity recognition in their personal research.

No prior knowledge of digital technologies or computer programming is required for this course but all students should plan to develop final projects or papers featuring original work related to one or more of the methods for natural language processing that we will employ.

Required Texts and Readings

(to be distributed in PDF format via Canvas)

Steven Bird, Ewan Klein, Edward Loper, *Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit* (O'Reilly 2009, website 2018)
http://www.nltk.org/book/

Dipanjan Sarkar, *Text Analytics with Python* (Apress/Springer, 2016)
https://link-springer-com.proxy.uchicago.edu/book/10.1007%2F978-1-4842-2388-8

All required readings for the course will be provided via the online Canvas platform at canvas.uchicago.edu . Any students without access to Canvas must inform the instructor so we can set up alternate methods for you to access the readings.

Further Reading and Digital Resources

Stanford University CS224n: Natural Language Processing with Deep Learning
http://web.stanford.edu/class/cs224n/

Paul Vierthaler's Stylometric PCA and Network Data Explorer
https://www.pvierth.com/pca

Course Plan and Policies

Monday and Wednesday sessions will mainly focus on reviewing the content of the assigned readings and include lectures on and discussions of specific topics. Friday sessions will be dedicated to open discussion and Python programming strategies, allowing for free-flowing, detailed and individualized discussions directly relevant to the week's assignments and topics.

Assignments

Weekly assignments will primarily be comprised of programming exercises in Python. The code and output is to be submitted to the instructor for evaluation by email unless otherwise directed.

A formal Final Project and Final Exam will be required of all students (see below).

The Final Exam will be comprised of multiple-choice questions and written responses and will be given at the time and date designated by the University's exam schedule. If for any reason you will not be able to take an exam as scheduled, you must gain prior approval from the instructor for alternate means to take the exam.

Final Project / Final Paper :

An initial proposal for the dataset(s) to be used in the final project will be due at the end of the second week, to be finalized by the end of Week 5.

Topic(s) for the final project/paper (or project white papers) are to be developed in consultation with the instructor and are to be **submitted in writing** (minimum of one paragraph in length) **by the end of Week 7. All projects/topics must have received written preapproval** (email is fine) **and will be set by the end of Week 8.**

Final projects should center on the analysis of a specific data source and include at least some of the methods we will cover and use in the course. Final projects that employ new and/or unique datasets and reach innovative conclusions will receive the highest scores.

A full written explanation of the scope and utility of the project, at least 3 pages in length (Times 12pt, double-spaced), will be required by the due date of the final project. All project coding and use of data sources will be closely reviewed, and the potential impact of the project will play a major role in its assessment. No group projects will be allowed.

All students will be given space and service units for analyses on Midway, the university's high-performance computing (HPC) cluster, depending on the needs and dependencies of each individual project, developed and maintained in consultation with the instructor. Students will be responsible for all administration and content management associated with their projects.

Students may choose to do a Final Paper instead of a Final Project. The paper must be between 10 and 15 pages in length (Times 12pt, double-spaced), and should provide detailed evaluations of and research into at least one digital resource, methodology and/or toolkit directly related to those covered in the course readings and discussions, and must include discussion of at least one programming toolkit and/or algorithm. Proper spelling, grammar and construction of your paper (thesis, argumentation, transitions, conclusions) will be strongly considered in its evaluation.

All Final Project Reports and Final Papers are due by midnight on the Friday of Exams Week. Penalties for late projects/papers will be assessed at a rate of one letter grade per day. If you will need an extension and/or to take a course grade of Incomplete, you must have received approval for this in writing (email is fine) from the instructor by midnight on Friday of Exams Week.

Attendance

The success of our course discussions depends upon your active participation, so your contributions are important to me. Please note that your attendance isn't enough to make this course successful; I expect that you will also participate regularly in class by sharing your own observations and ideas, comments and critiques.

Absences may be excused on account of documented illness, religious observances, participation in university-sponsored athletic events, and serious emergencies. Please let me know in advance if you will be missing class for any reason. You can miss up to 3 classes without penalty. After that, your final grade will be lowered one-third of a grade for each additional absence (A- becomes B+; B becomes B-, etc.).

Grading / Evaluation

Attendance and participation:	20%
Short projects and exercises (Assignments):	20%
Final exam:	20%
Final project or paper:	40%

Special Needs

Students with any form of special needs, physical, learning or otherwise, are welcome in my courses. It is University policy to provide, on a flexible and individualized basis, reasonable accommodations to students who have disabilities that may affect their ability to participate in course activities or to meet course requirements (see http://disabilities.uchicago.edu/). All students with disabilities should contact me to discuss their individual needs for accommodations.

Course Schedule

All readings are to be completed prior to the course time under which they are listed.

Students should also feel free to bring up any of the works in the "Further Reading" list during in-class discussions or in the Short Project reports.

"Bird" and "Sarkar" refer to the course texts listed above; all other readings will be made available in PDF format in the Canvas course system. All "Assignments" will be posted to the Canvas system and also distributed in class.

Week.Class	Themes / Topics	Readings & Assignments
1.1	Introduction & Syllabus	
1.2	What is Natural Language Processing?	*Bird : Preface*
1.3	NLTK, Python 3 and the Jupyter Notebook Introduction to HPC	*Bird : Chapter 1* ***Assignment 1***
2.1	Textual Sources and Formats 1: "What's in a Text?"	*Bird : Chapter 2*
2.2	Sources 2: APIs, Social Media, Web Scraping	*Web Scraping Handout*
2.3	Building your Corpus	***Assignment 2*** ***Dataset initial proposal due***
3.1	Tokenization, N-grams and *Scriptio continua*	*Bird : Chapter 3*
3.2	Stemming and Lemmatization, Synsets and Hypernyms	*Bird : Chapter 4*
3.3	Tokenizing your Corpus	***Assignment 3***

4.1	POS Tagging and Stopwords	*Bird : Chapter 5*
4.2	Text "Features" and TF-IDF Classification	*Bird : Chapter 6*
4.3	The "Words" in a "Text"	***Assignment 4***
5.1	Named Entity Recognition (NER)	*Bird : Chapter 7* *Sarkar pp.167-215*
5.2	Sentiment Analysis	*Sarkar pp.319-376*
5.3	What *Kind* of Text is it? (Machine Learning Approaches to Textual Data)	***Assignment 5*** ***Final dataset proposal due***
6.1	Topic Modeling Basics	*Sarkar pp.217-263*
6.2	Topic Modeling: Strengths, Weaknesses, Correlations	*TBD*
6.3	What's in a Topic?	***Assignment 6***
7.1	Stylometry & Stylometric Analysis	*Paul Vierthaler tutorials*
7.2	Dendograms, PCA scatterplots & *k-means*	*Paul Vierthaler tutorials*
7.3	Plotting the Text, Finding the Plot	***Assignment 7*** ***Final Project/Paper Topic Due***

8.1	Document Clustering and Word Vectors	*TBD*
8.2	Doc2vec, Word2vec	*TBD*
8.3	Advanced Vector Analyses	***Assignment 8*** ***Final Project / Paper Topic finalized***
9.1	Advanced Structures 1: Dependency Parsing	*Bird : Chapter 8*
9.2	Advanced Structures 2: Constituency Parsing	*Bird : Chapters 9 & 10*
9.3	The Worlds Beyond the Text	***Assignment 9***
10.1	Student Presentations	
10.2	Student Presentations	
	Final Exam	
	Final Papers / Projects Due	