

INLS 512
APPLICATIONS OF NATURAL LANGUAGE PROCESSING

Instructor: Fox
Email: foxm@unc.edu
Room: None
Schedule: Online Asynchronous
Office: Zoom, Greenlaw 505
Office Hours: 3:30-5:30pm on Wednesdays or by appointment

DESCRIPTION AND OBJECTIVES

Natural language processing (NLP) draws on mathematics, machine learning, linguistics, and computer science to make language, both spoken and written, computationally accessible and analyzable. In the first of two parts of this course, a practical introduction to NLP, you will learn to do essential NLP tasks using Python and to apply your newly acquired skills to a sampling of NLP applications. In the second part, you will survey a selection of NLP applications and acquire experience in describing the problems or tasks addressed by each application, the materials and methods used, and how the applications are evaluated. You will also develop critical skills in spotting limitations in studies and in imagining and articulating opportunities for enhancements or improvement to NLP applications. This course is targeted for both undergraduate and graduate students studying information science or related fields.

TEXTS

The texts for this course are available either online or on Canvas:

- [*Natural Language Processing with Python – Analyzing Text with the Natural Language Toolkit*](#) by Steven Bird, Ewan Klein, and Edward Loper (O'Reilly 2009, Web 2019)
- Handouts

The following are not required, but I recommend them as good textbooks on NLP, if you're interested in learning from them on your own:

- Eisenstein. *Introduction to Natural Language Processing*. MIT Press, 2019.
- Jurafsky and Martin. *Speech and Language Processing: An Introduction to Natural Language Processing, Computational Linguistics, and Speech Recognition* (2nd Edition). Pearson Education, 2014.

REQUIREMENTS (AND GRADE DISTRIBUTION)

Participate (10%)

Participation involves contributing a few times to the Canvas Piazza Q&A over the course of the semester. If there are no questions for you to answer on Piazza, you may post a few snippets of

code along with brief explanations of them that show you exploring the use of Python in NLP. The snippets don't have to be complicated. They can be extensions of your homework solutions, or they can simply show you working out an issue with your Python code. The idea is to help each other improve in coding ability.

Code (50%)

For Part One of the course, you will complete a set of Python programming exercises each week. You will upload your code and output in HTML format generated from Jupyter Notebook to Canvas Assignments.

To use Jupyter Notebook, you'll want to install Anaconda first. See here for instructions on how to install Anaconda and run Jupyter Notebook: <https://sparkbyexamples.com/python/install-anaconda-jupyter-notebook/>.

And here is a good list of keyboard shortcuts for Jupyter Notebook: <https://towardsdatascience.com/jupyter-notebook-shortcuts-bf0101a98330>

If you have problems with your code, especially if they involve syntax or other coding errors, please first try to figure them out on your own—doing so is the best way to master programming. You may also post questions to Canvas Piazza and answer each other's questions. Piazza is a great platform for Q&A exchanges. I will be monitoring our Piazza instance and answering questions as well. And, of course, you may ask me questions during office hours.

The programming exercises will be graded on fulfillment of the requirements—I'm not at all concerned with style, since NLP tasks can always be done in different ways.

Annotate (40%)

Each week of Part Two of the course, you will annotate one research article on an NLP application. Each annotation should be about 250 to 350 words and should provide, in addition to a concise summary of the source, a brief critique of it. Describe the problems or tasks addressed by each article, the materials and methods used, and how the application of the study was evaluated. Then, discuss any limitations you see in the study and potential opportunities for enhancements or improvements to the application it addresses. Your annotations will be graded on fulfillment of the requirements, quality of writing, and thoughtfulness.

GRADING SCALE

Undergraduate

A	> 92
A-	90-92
B+	88-89
B	83-87
B-	80-82

C+	78-79
C	73-77
C-	70-72
D+	68-69
D	63-67
D-	60-62
F	< 60

Graduate

H – High Pass

P – Pass

L – Low Pass

F – Fail

IN – Incomplete

HONOR CODE

Your full participation and observance of the University's Honor Code are expected. And you are not permitted to upload any content from this course, including recordings you may be permitted to make through Accessibility Resources & Service, to the web in any form, including but not limited to Chegg, Course Hero, Coursera, Google Drive, etc. If you post course content, you may be violating my intellectual property rights. If you post your own work from this course, you are allowing sites to profit from your intellectual property. In utilizing web sources to upload or download course content, you risk violating the University's Honor Code.

ACCESSIBILITY RESOURCES AND SERVICE

The University of North Carolina at Chapel Hill facilitates the implementation of reasonable accommodations, including resources and services, for students with disabilities, chronic medical conditions, a temporary disability or pregnancy complications resulting in barriers to fully accessing University courses, programs and activities. Accommodations are determined through the Office of Accessibility Resources and Service (ARS) for individuals with documented qualifying disabilities in accordance with applicable state and federal laws. See the ARS Website for contact information (<https://ars.unc.edu>) or email ars@unc.edu.

MISCELLANEOUS

I may make changes to the syllabus as necessary over the course of the semester.

SCHEDULE¹

Part One

¹ N stands for *Natural Language Processing with Python*.

The readings listed here for each week of Part One will help you with each week's Canvas assignment, which is due at the end of the last day of the week (Sunday). See the Code section above for general instructions.

Under Files on Canvas:

- See “Jupyter Lectures - First Eight Weeks” for the “Jupyter Lectures” for each week. Some of them come in multiple parts. The lectures mostly contain sample code with output and comments that will help you complete the weekly code assignments.
- See “Files Referred to in Code Assignments” for just that. In that folder, there are also Jupyter notebooks corresponding to Jupyter Lectures from Week 5 Part 2 onwards.

A * indicates a due date.

Week 1 (Jan. 9-15) – Introduction: NLP, NLTK and Jupyter

N: Preface, 1

Jupyter Lecture: Week 1 – Chapter1.html

*Code 1 due Jan. 15

Week 2 (Jan. 16-22) – Textual Sources and Formats

N: 2.1-2.4, 3.1-3.2

Jupyter Lecture: Week 2 Part 1 - Chapter2-Part1.html

Jupyter Lecture: Week 2 Part 2 - Chapter3-Sect1-2.html

*Code 2 due Jan. 22

If you want to use scraped text from the web for your assignment, look over:

Web scraping with BeautifulSoup.ipynb

Week 3 (Jan. 23-29) – Tokenization; Regex; Stemming; Lemmatization; Segmentation

N: 3.4-3.9

Jupyter Lecture: Week 3 - Chapter3-Sect4-9.html

*Code 3 due Jan. 29

Week 4 (Jan. 30-Feb. 5) – Synsets and -nyms; POS Tagging; N-Gram Tagging; Transformation-Based Tagging; Determining the Category of a Word

N: 2.5, 5.1-5.7

Jupyter Lecture: Week 4 Part 1 - Chapter2-Sect5-Chapter5-Sect1-2.html

Jupyter Lecture: Week 4 Part 2 - Chapter5-Sect3-5.html

*Code 4 due Feb. 5

Week 5 (Feb. 6-12) – Named Entity Recognition; Sentiment Analysis

N: Chapter 7

Jupyter Lecture: Week 5 Part 1 - Chapter7.html

Jupyter Lecture: Week 5 Part 2 - NER in NLTK and SpaCy.html

Jupyter Lecture: Week 5 Part 3 - Pattern and TextBlob for sentiment analysis.html

*Code 5 due Feb. 12

**Week 6 (Feb. 13-19) – Word Vectors and LSA; Cosine Similarity; TF-IDF and Clustering;
Advanced Vector Analyses**

[Introduction to Word Vectors](#)

Helpful optional material:

Word Vector readings-tutorials.pdf

Jupyter Lecture: Week 6 Part 1 - SpaCy Word Vector examples.html

Jupyter Lecture: Week 6 Part 2 - Gensim Word2Vec all_Shakespeare example.html

Jupyter Lecture: Week 6 Part 3 - TF-IDF Shakespeare.html

*Code 6 due Feb. 19

Week 7 (Feb. 20-26) – Topic Modeling

Underwood, [“Topic modeling made just simple enough.”](#)

Jupyter Lecture: Week 7 Part 1 - Topic Modeling example(gensim LDA + NLTK + SpaCy).html

Jupyter Lecture: Week 7 Part 2 - Topic Modeling (gensim LDA + NLTK +
SpaCy)_Shakespeare.html

Jupyter Lecture: Week 7 Part 3 - Topic Modeling evaluations.html

*Code 7 due Feb. 26

Week 8 (Feb. 27-Mar. 5) – Stylometrics: Hierarchical Cluster Analysis (HCA)

Dendrograms and Clustermaps; Principal Component Analysis (PCA) Scatterplots

Jupyter Lecture: Week 8 Part 1 - vierthaler_stylometry_HCA_example.html

Jupyter Lecture: Week 8 Part 2 - vierthaler_stylometry_PCA_example.html

*Code 8 due Mar. 5

Part Two

See the Annotate section above for general instructions.

Under Files on Canvas:

- See “Suggested Articles on NLP Applications” for a suggested article to annotate for each week. You may choose to write on articles you find on your own that fall under each category.

A * indicates a due date.

Week 9 (Mar. 6-12) – Spoken Dialogue Systems

*Annotation due Mar. 12

Week 10 (Mar. 13-19) – Spring Break

Week 11 (Mar. 20-26) – Generation

*Annotation due Mar. 26

Week 12 (Mar. 27-Apr. 2) – Deception

*Annotation due Apr. 2

Week 13 (Apr. 3-9) – Health and Medical

*Annotation due Apr. 9

Week 14 (Apr. 10-16) – Humanities

*Annotation due Apr. 16

Week 15 (Apr. 17-23) – Application of Your Choice 1

Choose any research article on an NLP application.

*Annotation due Apr. 23

Week 16 (Apr. 24-28) – Application of Your Choice 2

Choose any research article on an NLP application.

*Annotation due Apr. 28