NLP 2023: Natural Language Processing

Spring 2023

Instructor:	Alexander Kalinowski	**Time:**	Thursday 15:30 – 18:15
Email:	ajk437@drexel.edu	**Place:**	James Hall, Room 3114

Course Pages:

1. http://github.com/akalino/rowan-nlp

Office Hours: After class, or by appointment.

Main References: This is a restricted list of various interesting and useful books that will be touched during the course. You need to consult them during the course in order to develop the material required for the weekly presentations. The first text will be our main reference and is available in PDF format at the author's website. The second text is focused on pracrical NLP using the NLTK toolkit. The final reference will be useful when working with deep learning frameworks for NLP; in particular, we will focus on PyTorch.

- **SLP**: Dan Jurafsky and James H. Martin. *Speech and Language Processing (online draft)*, Pearson Prentice Hall, Upper Saddle River, N.J.
- **NLTK**: S. Bird, E. Klein, and E. Loper. *Natural Language Processing with Python: Analyzing Text with the Natural Language Toolkit*. O'Reilly, Beijing, 2009.
- **PYT**: Delip Rao, Brian McMahan. *Natural Language Processing with PyTorch*. 2019.

Secondary References: We will consider several bleeding-edge research papers in addition to the baseline text. These papers will favor applications of NLP to various domains:

- **R1**: Efficient Estimation of Word Representations in Vector Space. Tomas Mikolov, Kai Chen, Greg Corrado, Jeffrey Dean. https://arxiv.org/abs/1301.3781.
- **R2**: GloVe: Global Vectors for Word Representation. https://nlp.stanford.edu/pubs/glove.pdf.
- **R3**: Bag of Tricks for Efficient Text Classification. Armand Joulin, Edouard Grave, Piotr Bojanowski, Tomas Mikolov. https://arxiv.org/abs/1607.01759.
- **R4**: All-but-the-top: Simple and Effective Post-processing for Word Representations. Jiaqi Mu, Suma Bhat, Pramod Viswanath. https://arxiv.org/abs/1702.01417.
- **R5**: Text Understanding from Scratch. Xiang Zhang, Yann LeCun. https://arxiv.org/abs/1502.01710.
- **R6**: A Simple But Tough-to-Beat Baseline for Sentence Embeddings. Sanjeev Arora, Yingyu Liang, Tengyu Ma. https://openreview.net/pdf?id=SyK00v5xx.
- **R7**: Parameter-free Sentence Embedding via Orthogonal Basis. Ziyi Yang, Chenguang Zhu, Weizhu Chen. https://arxiv.org/abs/1810.00438.
- **R8**: Long Short-term Memory. Sepp Hochreiter, Jurgen Schmidhuber. https://blog.xpgreat.com/file/lstm.pdf.

- **R9**: Supervised Learning of Universal Sentence Representations from Natural Language Inference Data. Alexis Conneau, Douwe Kiela, Holger Schwenk, Loic Barrault, Antoine Bordes. https://arxiv.org/abs/1705.02364.
- **R10**: Attention is All You Need! Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, Illia Polosukhin. https://arxiv.org/abs/1706.03762.
- **R11**: BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. Jacob Devlin, Ming-Wei Chang, Kenton Lee, Kristina Toutanova. https://arxiv.org/abs/1810.04805.
- **R12**: DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter. Victor Sanh, Lysandre Debut, Julien Chaumond, Thomas Wolf. https://arxiv.org/abs/1910.01108.
- **R13**: RoBERTa: A Robustly Optimized BERT Pretraining Approach. Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, Veselin Stoyanov. https://arxiv.org/abs/1907.11692.
- **R14**: Improving Language Understanding by Generative Pre-Training. Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever. https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf.
- **R15**: Language Models are Unsupervised Multitask Learners. Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, Ilya Sutskever. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf.
- **R16**: BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension. Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, Luke Zettlemoyer. https://arxiv.org/abs/1910.13461.
- **R17**: Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer. Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, Peter J. Liu. https://arxiv.org/pdf/1910.10683.pdf.
- **R18**: Language Models as Knowledge Bases? Fabio Petroni, Tim Rocktaschel, Patrick Lewis, Anton Bakhtin, Yuxiang Wu, Alexander H. Miller, Sebastian Riedel. https://aclanthology.org/D19-1250.pdf.
- **R19**: Negated and Misprimed Probes for Pretrained Language Models: Birds Can Talk, But Cannot Fly. Nora Kassner, Hinrish Schutze. https://arxiv.org/abs/1911.03343.
- **R20**: A Review on Language Models as Knowledge Bases. Badr AlKhamissi, Millicent Li, Asli Celikyilmaz, Mona Diab, Marjan Ghazvininejad. https://arxiv.org/abs/2204.06031.
- **R21**: Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing. Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, Graham Neubig. https://arxiv.org/abs/2107.13586 and http://pretrain.nlpedia.ai/.
- **R22**: Learning bilingual word embeddings with (almost) no bilingual data. Mikel Artetxe, Gorka Labaka, Eneko Agirre. https://aclanthology.org/P17-1042/.
- **R23**: Unsupervised Machine Translation Using Monolingual Corpora Only. Guillaume Lample, Alexis Conneau, Ludovic Denoyer, Marc'Aurelio Ranzato. https://arxiv.org/abs/1711.00043.
- **R24**: On the Dangers of Stochastic Parrots: Can Language Models Be Too Big? Emily M. Bender, Timnit Gebru, Angelina McMillan-Major, Shmargaret Shmitchell. https://dl.acm.org/doi/10.1145/3442188.3445922

Objectives: This graduate-level, seminar style course will serve as an introduction to natural language processing, with a focus on demonstrating applications in modern data science. At the end of this course, students are expected to:

- Discuss standard ways of pre-processing text for use as features in machine learning algorithms;
- Detail and deploy modern pre-trained language models;
- Develop a codebase for using language models in practical settings;
- Discuss scalability issues, including the usage GPU machines;
- Identify application areas in the data sciences, including, but not limited to, domain adaptation, statistics and machine learning.

Prerequisites: A graduate-level understanding of probability, statistics, calculus, algorithms, and linear algebra is assumed. This course will demonstrate methods of NLP using Python; no prior experience with Python is **required**, but will be advantageous. PyTorch will be taught throughout the course and prior knowledge is not required. You should be comfortable working in an IDE, downloading and configuring python packages.

Tentative Course Outline: This outline is tentative, depending on how quickly/slowly we move through the material. You will also see that the tentative list of readings only extends through week 12– we will continue to identify *your* interests in NLP and add them to the final few weeks.

SLP Chapters 1 - 3	Week 1
NLTK Chapter 1, 2, 3, 6	Week 2
PYT Chapters 1, 3, 4, 5	Week 3
SLP Chapter 6 & **R1**	Week 4
R2, R3, R4, R5	Week 5
SLP Chapter 9 & **R6, R7, R8, R9**	Week 6
R10, R11, R12, R13	Week 7
R14, R15, R16, R17	Week 8
R18, R19, R20	Week 9
R21 Hands-on experiments with prompts	Week 10
SLP Chapter 13 & **R22, R23**	Week 11
SLP Chapter 14	Week 12

Grading Policy: Homework and quizzes (20%), Presentation and contribution to the course notes (40%), Final Project (40%).

Important Dates:

Homework 1	End of Week 5
Project Proposal	End of Week 7
Homework 2	In-class assignment, Week 10
Project Presentations	Final week

Class Policy:

- Regular attendance is essential and expected, periodic quizzes will be used to take attendance.
- Weekly readings should be read by ALL students, not just those presenting!

Academic Honesty: Lack of knowledge of the academic honesty policy is not a reasonable explanation for a violation.