Natural Language Processing for CogSci Research

Instructor: Marlene Staib

IMC/CogSci workshop, September 2018

E-mail: marlene.staib@gmail.com

Course Materials: **github**

Morning session: 9-12am
Office hour: 4-5pm

Afternoon session: 1-4pm
Room: 1485

Overview

This workshop gives an introduction to some of the Machine Learning techniques used for Natural Language Processing. By understanding and implementing some simple, exemplary models, participants will learn the basics of feature engineering, model selection and hyperparameter tuning. This will provide a foundation for studying and applying other approaches used in modern NLP. Discussions and labs will focus on the role of NLP for Cognitive Science, and the application of analyses to participants' own research.

Course Objectives

After the workshop, participants should be able to:

1. Understand the basic modelling approach(es) used in NLP
2. Implement a simple model for solving an NLP task
3. Critically evaluate choice of feature representation, type of model, hyperparameter selection
4. Understand the challenges and limitations for modelling **language**, in particular
5. Apply learned techniques to relevant questions in Cognitive Science research, e.g. to comparatively evaluate texts from different speaker groups

Organization

The main part of the workshop is split into two parts: seminars (lecture and discussion) and labs. The lab materials will be provided in the form of Jupyter/iPython Notebooks. See "Software" section below on how to get started with python and Jupyter.

To get the most out of the workshop, I will suggest materials (readings, blogposts, videos, labs/tutorials, quizzes) for you to engage with before (or in some cases after) the course. Materials are marked

[F] = foundational,
[C] = core, and
[A] = additional.

Please try to complete all of the core materials – there will only be a little bit of preparation, to familiarize yourselves with the topics and to introduce yourselves and your ideas to me. Foundational materials should help you gain a good background for what's going on in the lectures and labs. Feel free to skip them if they seem easy to you – they are meant as support, not as an additional burden. Additional materials are for those of you who have smelled blood and now want more. They are also meant to keep you busy, in case you happen to be one of the people who finishes each lab within half the allocated time.

I do not expect anyone to read anything, or watch any of the linked videos; but I have noticed that it often helps me solidify my knowledge if I hear/read the same content from at least 2-3 different sources. Also feel free to find your own content – e.g. by googling the keywords for each day. I found some really cool videos and blog posts that way, that were way more fun than the textbook.

Software

If you do not already have a working installation of Anaconda or Miniconda, please install Miniconda from the provided link. It doesn't really matter which version you have (if you do not have an installation yet, I recommend version 3.6), as long as you set up your environment as described below. I can give very limited support on problems relating to versions, operating systems and conflicting package installations, therefore I would like you to follow these instructions to set up a new conda environment for the course. If you are an expert and know you can make it work no matter what, you may do whatever you want ;)

Open a terminal (Mac/Linux) or command prompt (Windows). Run:

```
conda create -n nlp_workshop python=3.7
source activate nlp_workshop
conda install numpy pandas matplotlib seaborn ipykernel nltk scikit-learn
python -m ipykernel install --user --name nlp_workshop --display-name "Python (nlp)"
```

This creates a separate python environment for the course, which should not conflict with any other versions of python you have on your computer. You can activate and deactivate the environment with:

```
source activate nlp_workshop
source deactivate
```

Whenever the environment is activated, you can use that installation of python via the terminal/command prompt. Now, it's time to download the labs to your computer. If you have git installed, just run:

```
git clone https://github.com/MarleneStaib/NLPworkshop.git
```

Otherwise, open the workshop repository in your browser and download it as a zip file. Save it in your home directory. In your terminal, navigate to the folder/directory where you saved it, like this:

```
cd NLPworkshop/labs
```

Then type (with the environment still activated):

```
jupyter notebook
```

This should automatically open a browser window, which looks like this:

You can click on the labs to open them, but that's for later. You should be all set now. (To close the notebook, you can close the browser windows and then hit Ctrl+C in your terminal.) If you cannot manage to get there on your own, come see me **in my first office hour on Monday, 17th of September at 4pm.**

Timetable

Day 0: Preparation

Materials:

- Pre-course Survey: Please fill in the pre-course survey [C]
- Short intro video: NLP tasks and applications; overview [C]
- Set up your environment, see *Software*; if you run into trouble come see me during the first office hour before the course: **Monday, 17th of September at 4pm** [C]
- Optional: NLTK intro and tutorial [A]
- If you want a deeper understanding of the modelling approaches used in most of modern NLP, I recommend you to freshen up your linear algebra, probability theory and a tiny bit of calculus. [F]
- If you want an overview of NLP tasks and methods, I recommend the standard introduction "Speech and Language Processing" by Jurafsky and Martin (henceforth: J&M). Their 3rd edition draft is for free available online, and the most up-to-date, but incomplete. For specific topics, you may have to check out edition 2. There is also a whole course (corresponding roughly to the J&M book(s)) available for free on YouTube by Dan Jurafsky and Chris Manning from Stanford University. [A]

Day 1: Modeling in NLP; Naive Bayes

Morning: Lecture and discussion:

- Intro to NLP, reply to your comments from the online survey
- Modelling in NLP:
  - CogSci/Stats versus AI/Machine Learning
  - Inputs and outputs, feature representations for language
  - Modelling approaches: supervised/unsupervised, discriminative/generative, classification/regression
  - Hyperparameters
  - NLP - just another machine learning problem?
- First example: Naive Bayes for text classification
  - Some background in probability theory
  - Bayes' rule
  - The Naive Bayes Classifier: A generative model

Afternoon: Data Science in Python Lab

- `numpy, pandas, matplotlib, seaborn`
- *Note: If you are very familiar with these libraries, get bored during the lab or just finish everything early, you may get started on the Naive Bayes Lab linked under Day 2! :)*

Materials:

- Chapter 4 in J&M, ed. 3 [F]
- Lessons 6.1 to 6.9 from Jurafsky's lecture series on YouTube [F]

Day 2: Feature representations for language; Vector Semantics

Morning: Lecture and discussion:

- Possible representations for language; the problem with discrete representations
- Distributional hypothesis and vector semantics
- Measuring collocations and similarity: PPMI, TF-IDF, (PCA/LSA), cosine similarity

Afternoon: Naive Bayes Lab

- Train-validation-test split
- Building and evaluating a basic model
- Bonus: Improving the basic model: Feature engineering

Materials:

- Chapter 6 in J&M, ed.3 [F]

Day 3: Unsupervised Learning; k-means/GMM clustering

Morning: Lecture and discussion

- k-means clustering algorithm
- Choosing the number of clusters
- Gaussian Mixture Models

Afternoon: Vector Semantics Lab

- Turn some text into word vectors
- Experiment with different ways of creating vectors: TF-IDF, PPMI
- Measure cosine similarity between different word vectors
- Additive meaning of word vectors?
- Using word vectors for sentiment analysis

Materials:

- Lecture on k-means by Victor Lavrenko on YouTube [F]
- Lecture on GMMs by Victor Lavrenko on YouTube [A]

Day 4: Neural Networks

Morning: Lecture and Discussion

- High-level introduction to Neural Nets and their application in NLP
- Almost all of modern NLP is "deep" learning (using Deep Neural Networks, DNNs). This has achieved some amazing results, but there are still some challenges specific to language that are unlikely to be solved with DNNs.
- We can discuss issues such as: What types of networks reflect what features of language?

Afternoon: Unsupervised Learning Lab

- Clustering word vectors with k-means and GMMs
- Defining the "optimal" number of clusters empirically
- Optional: Hierarchical Clustering
- *Note: If you are done with all the labs early, you could have a look at this PyTorch tutorial and start working on Neural Nets :)*

Materials:

- Intro to Neural Networks by 3blue1brown. [A]

- Skip gram: Learning vector semantic representations with a neural network (so-called "word embeddings"). [A]
- This is way beyond what is covered here, but if you are convinced that NLP is awesome, feel free to check out Stanford's course on Natural Language Processing with Deep Learning, which is fully available on youtube. [A]
- Once you have a bit of an understanding of Deep Learning and NLP, you may want to check out this amazing PyTorch tutorial, to implement your own model. [A]