What is Natural Language Processing (NLP)?

From Wikipedia: "Natural language processing (
of linguistics, computer science, information en
artificial intelligen en
computers and hu to
program computers to process and analyze lar
natural language data."

What is Computational Linguistics (CL)?

From the ACL website: "Computational linguistics is the scientific study of language from a computational perspective. Computational linguists are interested in provid l models of various kinds of linguistic phenomena. T may be "knowledg iven" ("statistical" or "e istics is in some cases motivated from a scientific persp trying to provide a computational explanatio linguistic or psycholinguistic phenomenon motivation may be more purely technological in that one wants to provide a working component of a speech or natural language system. "

What is the relation between CL and NLP?

- Most of the time the two terms are used interchangeably.
- In practice, some researchers are more inter
  computational models as an explanation of s
  phenomena (ted
  with a tree?”). w
  working system or an NLP application (a ion
  system (MT), a dialogue system).
- There are more and more of the latter types
  applications are put to everyday use. Have you ordered
  anything with Alexa?

NLP Applications

- Sentiment, opinion, emotion analysis
- Information Extraction, Knowledge Acq

- Question An
- Machine Tra

- Text summarization

- Spoken Language Understanding, Di
- Many others

Sentiment Analysis

Information Extraction

Machine Translation

Machine Translation

Machine Reading

Machine Reading

Question Answering

Text Summarization

Dialogue Systems

NLP Tasks

Many of these problems are complex and cannot be solved with a single model. So they are decomposed into smaller self-contained problems that can be solved individually and then ch into a pipeline system. These problems are called in NLP convention.

Take Information

- Named Entit
- Coreference resolution
- Relation Extraction
- Knowledge base population

Pipeline systems are susceptible to **error propagation**. Anytime you can combine individual tasks and do **joint inference**, you can usually improve system performance.

NLP Tasks are often influenced by linguistic conceptions

Linguistic layer	NLP task
Morphology	Word tokenization, orpho-logical analysis
Syntax	dependen-
Semantics	Semantic role lab resenta- tion parsing
Discourse	Discourse parsi
Pragmatics	Dialogue act tagging

Formal characterization of NLP tasks

- Simple classification problems
  - Sentiment/opinion/emotion analysis, text classification, word sense disambiguation (WSD), etc...
- Sequence labeling problems
  - Tokeni de
    switchi ion (MWE)
    detecti
- Problems that can be modeled as trees and ing
  - Syntactic (dependency and constit
    representation parsing
- Sequence-to-sequence problems
  - Machine Translation, Text Summarization, dialogue systems(?)

Steps in developing an application

Supposed you are asked (or wanted) to develop an NL application, you need to think about:

- How do I deco ven the current st
- For each task, what is the most appropriat method for each of the tasks?
- What type of training data should I create (purchase, license)?

General formulation of Learning

Many NLP problems ca
optimzation:

$$\hat{\boldsymbol{y}} = \text{argmax}_{\boldsymbol{y} \in \mathcal{Y}} \boldsymbol{\Psi}(\boldsymbol{x},$$

where,

- $\boldsymbol{x}$ is an element of a set $\mathcal{X}$
- $\mathbf{y}$ is an element of the set $\mathcal{Y}(\boldsymbol{x})$
- Ψ is a scoring function or a **model**, which maps from the set $\mathcal{X} \times \mathcal{Y}$ to real numbers
- $\boldsymbol{\theta}$ is a vector of parameters for $\boldsymbol{\Psi}$
- $\hat{\boldsymbol{y}}$ is the predicted output, which is chosen to maximize the scoring function

Learning and search

- Search is the procedure of finding the output $\hat{y}$ that gets the best score with respect to the input $\boldsymbol{x}$ by computing the argmax of the scoring function Ψ
  - The search can be simple if it's a matter of findi label am)
  - Or it need e best part
- Learning is the process of finding the para $\boldsymbol{\theta}$.
  - This is done by optimizing some functi labeled data in a training process
  - The parameters are usually continuous, and learning algorithms generally rely on **numerical optimization** to identify vectors of real-valued numbers

Learning Methods for Simple Classification

- Supervised Learning
  - Generative
    - Naive Bayes
  - Discriminative
    - linear models: Logistic Regression, P
      V
    - N earning,
      m
- Unsupervised Learning
  - EM-based algorithms (backward side), clustering algorithms (K-means, EM, Hierarchical)
- Semi-supervised methods
- Search is usually trivial, and involves finding the label that gets the highest score.

Sequence Labeling methods

Sequence labeling methods can be viewed as classification combined with a search algorithm.

- Supervised Learning
  - Generative
    - Hidden Markov Models (HMM): Nai th th
  - Discri
    - li ogistic Regression combined with the Vi eptron combined with the Viterbi Algori
    - Non-linear models: LSTM-CRF, a form of RNN combined with a search algorithm
- Unsupervised Learning
  - Backward-forward, a form of EM algorithm for sequences
- semi-supervised methods

Tree-based learning algorithms

Tree-based methods c d with a search algorithm

- Supervised Learning
  - Generative parsing models
    - Naive Bayes combined with the CKY al nt parsing

  - Discri
    - P dency parsing

    - Perceptron or logistic regression com for constituent parsing

  - Non-linear models:
    - LSTM combined with CKY
- Unsupervised Learning
  - Inside-outside, a form of EM algorithm for trees
- semi-supervised methods

Learning and linguistic knowledge

- The relative impo e has been a recurring topic of debate.
  - "Every time I fire a linguist, the performance of our speech recognition system goes up."
- Linguistic knowledge figures prominently systems
- Statistical sy eatures linguisticall
- Deep learning methods enabled end-to
  - Particularly effective in areas like M scale training sets
  - "Natural language processing from scratch"
  - Model architectures are still inspired by linguistic theories
- The debate is far from being settled. Linguistic knowledge is particularly needed in problems that require "deep understanding".

What type of math is needed

- Probabilities
  - The output of a classifier is often expressed in terms of a probabilistic distribution: "This review has a 82% probability of being positive, and a 18% probability of be
  - For some models (e.g., naive Bayes), the p are in the form of probabilities
- Calculus: kno functions
  - The most basic form of machine learni ed on gradient descent/ascent to achie minimums
- Linear algebra: increasingly, you have to manipulate vectors and matricies (or "tensors")
- Some of you may not have calculus. We plan to have some tutorials on this topic.

Where do you need linguistics?

- Most of the NLP tas
  intuitive.
  - The part speech of a word: verbs, nouns, adjectives
  - Named entities: person, organization, g
    geo-political entities, etc.
  - Reviews: This is a positive review
- Some linguis c
  training
  - Syntactic trees (constituent or depe
- When breaking down an application int
  you can solve with machine learning, you need a good understanding of different layers of analysis: morphology, syntax, semantics, discourse and dialogues, etc.
- You also need good linguistic intuition to come up with good features or architectures in your statistical model

What kind of programming skills do you need?

- Python, Python, Python!
- Python libraries:
  - numpy part of Tensor
  - We are going to use PyTorch for some of o
- We are hold tutorials on PyTorch for peop this during recitation.

Course requirements

- Prerequisites
  - CS114 or permission by instructor (talk to
    stars you should take the course)
  - Programming experience: familiar with Python and Python packag
  - Statisti nd ability to pick up some linguistics
- Disciplines Relevant to NLP
  - Linguistics
  - Computer science, Artificial Intelligence
  - Machine learning

Textbooks

- Textbooks
  - Required: Introduction to Natural Language Processing by Jacob Eisenstein (pre-publication vers from https://github.com/jacobeise class/b es.pdf)
  - Recom ex J. Smola: accessed from https://d2l.ai)
- Supplemental online material
  - PyTorch tutorials: https://pytorch.org/tutorials/beginner/deep_learning_60min_blitz.html
  - PyTorch Documentation: https://pytorch.org/docs/stable/index.html

Perusall

- I plan to experiment with Perusall: https://a /
- Perusall allows users to highlight specific are
  document to e
- When possib
  each the class so that you can enter comme
  I can then address those questions during t
- To access the course, enter the access code: XUE-XPWJB

Course work

- Participation: 5
  - Credits for active participation in class discussions, and for asking questions.
- 4 Projects (50%)
  - Projects are more open-ended, though st for some ite-up.
- 3-4 Assignm
  - Programming assignments for well
- 1 final Quiz (15%)
  - The quiz tests important concepts covered in the course.
- **Academic integrity**: You should finish homework assignments, exams, and project reports on your own unless a project is explicitly stated as a collaborative project. Late projects are subject to grade deduction.