Introduction to Computational Linguistics

Modeling Language for CompLing & NLP

Course overview

Olga Zamarëva (2019)
University of Washington
April 2, 2019
But first: Emergency procedures

https://www.washington.edu/uwem/plans-and-procedures/uw-emergency-procedures/

Instructor’s info

- Olga Zamarraeva (preferred address: Olga)
  - a graduate student at the Linguistics dept.
  - that means...
- TA: Sara Ng
- Office: GUG407
- Office hours on the website
- Questions: please use Canvas discussion board for non-confidential stuff
Prerequisites and alternatives

- **Prerequisites**
  - Basic linguistic theory (LING200)
  - Basic programming skills are a big plus (e.g. CSE142)
  - Basic probability
  - Maturity when it comes to dealing with open source and academic software

- **Alternatives: pure NLP perspective**
  - CSE447 (NLP)
  - LING570/572 (and 571)
  - Require strong probability, programming, and basic machine learning background

- **Wait list**
  - Email me!
A survey course on computational linguistics...

- CompLing and NLP are both relatively young fields
- ...which change really fast
- ...which used to be one field (community) in the 90s
- ...which are two different fields (communities) today
- We will have to balance surveying topics relevant to state-of-the-art with studying fundamentals
  - Not possible to study state-of-the-art directly
  - ...it relies on too many prerequisites which are not yet part of e.g. standard math, programming, or social science curriculum
  - So we will sometimes spend time on things which may seem out of date
  - Everything we will study is relevant to state-of-the-art in some way
    - ...but you are unlikely to encounter them in new ACL paper titles
Course Logistics

- Canvas: https://canvas.uw.edu/ (CSE472 Sp19)
- Public website: http://courses.washington.edu/ling472/
- Lecture T/Th, Section F; 2-3 guest lectures (TBA)
- Readings: Jurafsky & Martin, can use 3rd ed. when available
- 5 Homeworoks (due on Fridays; HW5 is in two parts)
- Midterm on 5/2 (open notes, closed book, closed computers)
- Project (1st milestone: 4/26)
- Write ups weigh 50% of the homework grade (60% for project)
- For some assignments, showing effort will get you a long way
- Something is due every week... but this makes each assignment lighter...
Course Logistics: Reading Questions

- Graded Canvas discussions
- Ask or answer a question about the reading that is due next day
- Due night before lecture
- First one is due tomorrow!
- If you can’t complete all the readings, you can still formulate a question about the part you did complete!
- Some readings are shorter, some are longer; in order to complete the readings for the week, start early.
  - e.g. do not leave all Thursday reading for Wednesday
- Your question/answer should be thoughtful
- Worth 5% of your final grade
Course Logistics: How to approach the readings

- Some of the reading is very detailed, technical, and can be hard
- Identify the main idea/topic of the section/paragraph
- Try to relate this idea/topic to the bigger picture: Why is it important? How/where is it used?
- Ask questions
- Often it helps to reread some of the more confusing sections several times
- However, do not get too boggled; you do not have to understand every single word every single time
- You need to understand enough to complete the homeworks, so, refer to them as you read
- Some algorithms/concepts, you will need to know in detail; them being mentioned in lecture is your cue!
Course Logistics: Homeworks

- Depending on your background: some parts will be easy, some will be hard
- Everyone will find some easy and some hard parts in this class
- Assignment 5 is hard (for everyone); start as early as possible
- Write ups are worth 50%; please take it seriously!
- In your write up, show what you learned while completing the assignment; discuss challenges
- If you haven’t completed the assignment fully, the write up is your chance to earn maximum partial credit
- If you did complete it, you still must submit a coherent and thoughtful write up in order to earn full credit.
- All submitted code must run on patas so, please test!
- Finally, homeworks include reading detailed homework descriptions
  - ...sorry?
Course Logistics: Coding

- Most programs used in class are written in python3
- You can edit python programs using any editor, but
- ...an IDE such as PyCharm will let you step through your programs to debug them!
- PyCharm community edition (free download)
  - https://www.jetbrains.com/pycharm/download/
- There are many (presumably good) tutorials on basic programming in python
Course Logistics: Project

- http://courses.washington.edu/ling472/final_project.html
- Error Analysis on a NLP tool of your choice*
  - *a peer reviewed tool that you can download along with the dataset and run
- You must show understanding of what EA is
  - First clue (to be repeated): it is not the same as evaluation!
  - Instead, it is a methodical analysis of the items the system gets wrong, reasons and implications of that
- Groups (2-3 people) must consist of people with diverse background
- Final write up due 6/13 (instead of the final exam)
- In the write up, offer a meaningful categorization and reflect on the errors using what you learned in class
- Worth 30% of your final grade
Questions?
Computational Linguistics & NLP

CompLing == NLP?  CompLing ⊃ NLP?  CompLing ∩ NLP = ∅?
Computational Linguistics & NLP in LING/CSE472

CompLing == NLP?  CompLing ⊃ NLP?  CompLing ∩ NLP = ∅?

▶ Goal: A survey of the space that both CompLing & NLP live in

▶ Linguistic perspective

▶ Use working definitions for CompLing, NLP
NLP and CompLing: Poll

https://pollev.com/olgazamaraev657
Working Definitions

- Computational Linguistics & Natural Language Processing
  - Analyzing language material to solve tasks and answer questions

- Natural Language Processing (NLP)
  - Focus: existing data and algorithms
  - (Learning “something about the world”?)
    - speech recognition, dialog systems
    - machine translation
    - text classification, summarization, sentiment analysis
    - text and speech generation

- Computational Linguistics
  - Focus: study of language with computational means
    - documenting understudied languages
    - rigorous testing of syntactic theories
    - building typological databases
    - semantic ontologies
    - modeling human linguistic behavior
Modeling language

“In any mathematical theory about an empirical domain, the phenomena of interest are modelled as mathematical structures, certain aspects of which are conventionally understood as corresponding to observables in the domain”

Pollard & Sag 1994

▶ If we want to do anything with language, we need a way to represent language.

▶ What does it mean to model language?
  ▶ Do linguists need to model language?
  ▶ Do computer scientists need to model language?

▶ How close is a model to the real thing?
  ▶ Does it matter to linguists?
  ▶ Does it matter to computer scientists? To engineers?
Modeling language: statistical vs. symbolic methods

- Statistical methods involve training a model on a body of data so it can predict the most probable label/structure/etc for new data.

- Symbolic methods involve knowledge engineering, or hand-coding of linguistic knowledge which is then applied to tasks.

- Statistical methods provide robustness, symbolic methods precision.

- Statistical and symbolic methods can be and often are combined in both NLP and CompLing.

- Which methods are better?
Modeling Language: where to start?

- We can interact with the computer in several ways:
  - write or read text
  - speak or listen to speech

- Computer has to have some way to represent
  - text
  - speech

- Next, we can talk about modeling:
  - sounds of language (phonetics & phonology)
  - words (morphology)
  - sentence structure (syntax)
  - meaning (semantics, pragmatics...)

...
Relating texts to languages

- Texts represent a *fragment* of a language, in terms of grammar and lexicon
- Texts often represent *standard varieties*
  - What about speech?
- What would it mean to limit data to standard varieties?

picture of Dirk Hovy giving his talk: https://www.youtube.com/watch?v=gcfKvy4NrOE
Questions?
Topics we will cover

- only *roughly* in order...
- all are important but we don’t have time to focus equally on all
- no assignment on ASR but you can choose an ASR or TTS related project
- machine learning: a general overview
- focus on linguistic structure
Topics: Regular Expressions

- why do we start with RegEx?
  - they are the first tool people tend to use in practice when dealing with text
  - are part of formal language theory
- sequence of characters that defines a (search) pattern
- colou?r
  - (matches both *color* and *colour*)
- While directly relevant to formal languages, RegEx are very practical, so, practice:
- https://regex101.com/
Finite State Automata

- describe *regular languages* (sets of strings)
- are “equivalent” to RegEx (a RegEx and an FSA can describe the same “language”)
- **finite**: have a fixed number of states
- At any point, have access only to the current state, and can travel to the next state given an input
- Do not have “memory” (no stack!), and cannot model recursion

*colou?r*
Regular Languages

...are part of the Chomsky Hierarchy:

Topics: Morphology and Phonology

**Morphology**: Verbal position classes in Finnish:

- **stem** → **voice** → **tense** → **person** → **prtcl**

**Phonology**: Palatalization in Russian:

- $C \xrightarrow{c:c^i} [i]$
Finite State Transducers

...describe regular relations: sets of pairs of strings
\(\text{cat+s} : \text{cat+N+PL}\)
Topics: Sounds as language

We want to be able to encode any spoken language
(Btw: the $\approx 7000$ spoken languages, $\approx 3000$ have do not have
writing systems.)

- The International Phonetic Alphabet (IPA)
- Interactive example chart: http://web.uvic.ca/ling/resources/ipa/charts/IPAlab/IPAlab.htm

picture from: https://home.cc.umanitoba.ca/~robh/howto.html
Applications of speech encoding

Many applications for encoding speech, e.g.:

- Spoken dialog systems, i.e. speak with a computer (and have it speak back).
- Helping speech pathologists diagnose problems

Mapping sounds to symbols (alphabet), and vice versa:

- **Automatic Speech Recognition (ASR)**: sounds to text
- **Text-to-Speech Synthesis (TTS)**: texts to sounds
- **Dialog systems**: a combination
Topics: Machine Learning

- A large family of algorithms that train as they run
- Learn to output predictions for previously unseen* data
  - But there is “no free lunch” (see Mitchell’s paper)
- Require data in a numeric vector format
Machine Learning: Variance and Bias

Prediction error as a function of model complexity: train v. true error

\[
\text{error}_{\text{train}}(w) = \frac{1}{N_{\text{train}}} \sum_{j=1}^{N_{\text{train}}} \left( t(y_j) - \sum_{i} w_i h_i(x_j) \right)^2
\]

\[
\text{error}_{\text{true}}(w) = \int_t \left( t(x) - \sum_{i} w_i h_i(x) \right)^2 p(x) dx
\]

▶ A key problem in ML, scribbled on this slide by Carlos Guestrin's own hand!
Evaluation and Error Analysis

needed for the Final Project (and for everything computational!)

NB: Error analysis goes **beyond** precision and recall

pic by Walber - Own work, CC BY-SA 4.0, https://commons.wikimedia.org/w/index.php?curid=36926283
Statistical beginnings: N-grams and Language Models

▶ A basic technique to turn a document into a vector of numbers

Material based upon chapter 5 of Jurafsky and Martin 2009
Topics: Syntax

```
S
 NP      VP
    /
   l saw NP
      the astronomer PP
          with the telescope

S
 NP      VP
    /
   l saw NP PP
      the astronomer with the telescope
```
CKY demo: http://lxmls.it.pt/2015/cky.html
What is behind the trees?
If you want get a sense what it is about:
read *The Linguistics Wars* by R. Harris

picture from:
Grammar engineering

- In linguistic terms, creating formal, machine- and human-readable grammar descriptions
- Grammars map strings to syntactic and semantic representations
- *Linguistic* grammars are also language descriptions grounded in linguistic theory
- Demo: http://erg.delph-in.net/logon
Topics: Semantics

- Formal semantics is couched in formal logic
- “Every dog chases some cat”
- $\forall x \text{ dog}(x) \exists y \text{ cat}(y) \text{ chase}(x,y)$
- $\exists y \text{ cat}(y) \forall x \text{ dog}(x) \text{ chase}(x,y)$

picture source: https://imgur.com/a/XBHub
Computational Semantics

- **Semantic parsing**: Mapping a string to a semantic representation
- ...which may take different forms, e.g. AMR (abstract meaning representation, Banarescu et al., 2013)
Computational Semantics

- **Semantic parsing:** Mapping a string to a semantic representation
- ...which may take different forms, e.g. AMR (abstract meaning representation, Banarescu et al., 2013)

Every dog chases some cat.

(1)

<table>
<thead>
<tr>
<th>mrs</th>
<th>LTOP h1</th>
<th>INDEX e3</th>
</tr>
</thead>
<tbody>
<tr>
<td>RELS</td>
<td>every_q.rel</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>LBL h4</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>ARG0 x6</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>RSTR h7</td>
<td>...</td>
</tr>
<tr>
<td></td>
<td>BODY h5</td>
<td>...</td>
</tr>
<tr>
<td>HCONS</td>
<td>qeq</td>
<td>qeq</td>
</tr>
<tr>
<td></td>
<td>HARG h1</td>
<td>HARG h7</td>
</tr>
<tr>
<td></td>
<td>LARG h2</td>
<td>LARG h8</td>
</tr>
</tbody>
</table>
Vector space semantics

- Latest technique: Word embeddings
- Words are converted to vectors of numbers, starting randomly
- (cf. N-grams are documents converted to vectors of numbers by simple counting)

picture credit:
Topics: Deep learning

- One of the most successful techniques used in NLP in recent years.
- Surpasses human performance at some tasks
- Allows to learn features automatically (cf. classical ML)
- Requires training with millions of parameters (=expensive and not always feasible)
- Discussions about interpretability of DL results
- If interested, check out Joe Pater’s position paper in *Language* vol. 95 and 6(!) responses to it

picture from: https://mapr.com/blog/demystifying-ai-ml-dl/
Topics: Ethics and NLP

▶ What does NLP research do and Why?
▶ What is the researcher’s responsibility?

pictures from Dirk Hovy’s talk: https://www.youtube.com/watch?v=gcfKvy4NroE
What you need to know

▶ This class is a survey of how and why language data can be analyzed by computers
▶ It will be easy at times and hard at times, for everyone (in a different way)
▶ There is an error analysis final project (30%), completed in groups of 2-3
▶ There is no final exam; in-class midterm is 20% of the grade
▶ Readings are graded via reading questions (5%)
▶ Write ups are worth 50% of every assignment
▶ First RQ is due tomorrow (I am sorry!)
   ▶ For this time, it will be understandable if you don’t complete all the readings on time
   ▶ In general, start reading (as well as all assignments) ahead of time