Accelerated Natural Language Processing
Lecture 1
Introduction

Sharon Goldwater
(based on slides by Philipp Koehn)
Other lecturer: Henry Thompson

19 September 2016
What is Natural Language Processing?
who is the first Indian president

Rajendra Prasad
The 1st President of India

List of Presidents of India - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/List_of_Presidents_of_India
The President of India is the head of state and first citizen of India. The President is also the Commander-in-Chief of the Indian Armed Forces. Although the ... Zakir Hussain - Rajendra Prasad - VV Giri - R. Venkataraman

Rajendra Prasad - Wikipedia, the free encyclopedia
en.wikipedia.org/wiki/Rajendra_Prasad
listen (help·info); 3 December 1884 – 28 February 1963) was the first President of the Republic of India. An Indian political leader, lawyer by training, Prasad ...
What is Natural Language Processing?

Applications
• Machine Translation
• Information Retrieval
• Question Answering
• Dialogue Systems
• Information Extraction
• Summarization
• Sentiment Analysis
• ...

Core technologies
• Morphological analysis
• Part-of-speech tagging
• Syntactic parsing
• Named-entity recognition
• Coreference resolution
• Word sense disambiguation
• Textual entailment
• ...

Sharon Goldwater ANLP Lecture 1
NLP or computational linguistics?

- Scientist vs. engineer
- Explaining language vs. building applications
- Insight vs. empirical results
This Course

Linguistics
- words
- morphology
- parts of speech
- syntax
- semantics
- discourse

Computational methods
- finite state machines (morphological analysis, POS tagging)
- grammars and parsing (CKY, statistical parsing)
- probabilistic models (HMMS, PCFGs, MaxEnt models)
- vector spaces (distributional semantics)
- lambda calculus (compositional semantics)
This is a simple sentence
This is a simple sentence

be
3sg
present

WORDS
MORPHOLOGY

Sharon Goldwater  ANLP Lecture 1
### Parts of Speech

<table>
<thead>
<tr>
<th>PART OF SPEECH</th>
<th>WORDS</th>
<th>MORPHOLOGY</th>
</tr>
</thead>
<tbody>
<tr>
<td>DT</td>
<td>This</td>
<td>be</td>
</tr>
<tr>
<td>VBZ</td>
<td>is</td>
<td>3sg</td>
</tr>
<tr>
<td>DT</td>
<td>a</td>
<td>present</td>
</tr>
<tr>
<td>JJ</td>
<td>simple</td>
<td></td>
</tr>
<tr>
<td>NN</td>
<td>sentence</td>
<td></td>
</tr>
</tbody>
</table>
This is a simple sentence

This be 3sg present

be 3sg present
Semantics

This is a simple sentence

be 3sg present

SYNTAX

PART OF SPEECH

WORDS

MORPHOLOGY

SEMANTICS

Sharon Goldwater

ANLP Lecture 1
This is a simple sentence.

But it is an instructive one.
Why is Language Hard?

- Ambiguities on many levels, need context to disambiguate
- Rules, but many exceptions
- Language is infinite, cannot see examples of everything
Data: Words

Possible definition: strings of letters separated by spaces

• But how about:
  – punctuation: commas, periods, etc are normally not part of words, but others less clear: high-risk, Joe’s, @sloppyjoe
  – compounds: website, Computerlinguistikvorlesung

• And what if there are no spaces:
  伦敦每日快报指出,两台记载黛安娜王妃一九九七年巴黎死亡车祸调查资料的手提电脑,被从前大都会警察总长的办公室里偷走.

Processing text to decide/extract words is called tokenization.
### Word Counts

Out of 24m total word tokens (instances) in the English Europarl corpus, the most frequent are:

<table>
<thead>
<tr>
<th>any word</th>
<th>Frequency</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>1,698,599</td>
<td>the</td>
</tr>
<tr>
<td></td>
<td>849,256</td>
<td>of</td>
</tr>
<tr>
<td></td>
<td>793,731</td>
<td>to</td>
</tr>
<tr>
<td></td>
<td>640,257</td>
<td>and</td>
</tr>
<tr>
<td></td>
<td>508,560</td>
<td>in</td>
</tr>
<tr>
<td></td>
<td>407,638</td>
<td>that</td>
</tr>
<tr>
<td></td>
<td>400,467</td>
<td>is</td>
</tr>
<tr>
<td></td>
<td>394,778</td>
<td>a</td>
</tr>
<tr>
<td></td>
<td>263,040</td>
<td>I</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>nouns</th>
<th>Frequency</th>
<th>Token</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>124,598</td>
<td>European</td>
</tr>
<tr>
<td></td>
<td>104,325</td>
<td>Mr</td>
</tr>
<tr>
<td></td>
<td>92,195</td>
<td>Commission</td>
</tr>
<tr>
<td></td>
<td>66,781</td>
<td>President</td>
</tr>
<tr>
<td></td>
<td>62,867</td>
<td>Parliament</td>
</tr>
<tr>
<td></td>
<td>57,804</td>
<td>Union</td>
</tr>
<tr>
<td></td>
<td>53,683</td>
<td>report</td>
</tr>
<tr>
<td></td>
<td>53,547</td>
<td>Council</td>
</tr>
<tr>
<td></td>
<td>45,842</td>
<td>States</td>
</tr>
</tbody>
</table>
Word Counts

But there are 93638 distinct words (types) altogether, and 36231 occur only once! Examples:

- cornflakes, mathematicians, fuzziness, jumbling
- pseudo-rapporteur, lobby-ridden, perfunctorily,
- Lycketoft, UNCITRAL, H-0695
- policyfor, Commissioneris, 145.95, 27a
Plotting word frequencies

Order words by frequency. What is the freq of $n$th ranked word?
Plotting word frequencies

Order words by frequency. What is the freq of $n$th ranked word?
Rescaling the axes

To really see what’s going on, use logarithmic axes:
Zipf’s law

Summarizes the behaviour we just saw:

\[ f \times r \approx k \]

- \( f \) = frequency of a word
- \( r \) = rank of a word (if sorted by frequency)
- \( k \) = a constant
Zipf’s law

Summarizes the behaviour we just saw:

\[ f \times r \approx k \]

- \( f \) = frequency of a word
- \( r \) = rank of a word (if sorted by frequency)
- \( k \) = a constant

Why a line in log-scales?

\[ fr = k \Rightarrow f = \frac{k}{r} \Rightarrow \log f = \log k - \log r \]
Linguistics and Data

• Data
  – looking at real use of language in text
  – can learn a lot from empirical evidence
  – but: Zipf’s law: there will be always instances that are rarely seen

• Linguistics
  – build a better understanding of language structure
  – linguistic analysis points to what is important
  – but: many ambiguities cannot be explained easily
Course organization

- Lecturers: Sharon Goldwater, Henry Thompson
- Lectures, 3 per week:
  - Mondays/Thursdays 1510-1600, Teviot LT in Med Sch Teviot
  - Tuesdays 1110-1200, F.21 in 7 George Sq
- Weekly, in alternate weeks (normally Tue/Wed but this week on Wed/Thu):
  - 1.5 hr lab for exploring data and developing practical skills
  - 1 hr tutorial for working through maths and algorithms
  - See website for times/locations.
- Labs will be done in pairs; tutorial work can be done with whomever you choose.
Course materials

• All info online at
  http://www.inf.ed.ac.uk/teaching/courses/anlp/


• Labs, assignments, code: all on web page.
Assessment

• Three assignments, worth 10% each (30% total)
  – require some programming, but assessed on explanations and “lab-report” style write-ups.
  – You may (and are encouraged to) work in pairs.

• Exam in December, worth 70% of final mark.
  – short factual answers, longer open-ended answers, problem-solving (maths, linguistics, algorithms).
British higher education system

- Main principle: self-study guided by non-assessed work (some of it used for formative feedback), final assessed exam.

- Do not expect to learn everything just by sitting in lectures and tutorials! Most of your time should be in self-study:
  - Labs: intended to be done during scheduled lab times, but you may wish to look over them in advance (or revise after).
  - Tutorial sessions: work through problems in advance, and bring questions. Sessions can provide feedback on how well you understand the material.
  - Other: reading textbook, working through examples, seeking out online materials, group study sessions.
Background needed for this course?

- Know or currently learning Python.

- Background in Linguistics and prepared to learn maths (mainly probability) and algorithms.

- Background in CS and prepared to learn linguistics (and maybe maths).
Advice/warnings

• Students with little programming/maths: you can do it, but it will be very intensive.
  – Find study partners, start work early.
  – Pair up with a computer scientist.

• Students with programming but little maths or weak English: you can do it, but it will be very intensive.
  – Find study partners, start work early.
  – Pair up with a linguist or someone with stronger English.

• Students with strong programming/maths/machine learning: likely somewhat easier for you, but plenty of scope to learn and challenge yourself.
What this course is, and isn’t

This course is a fast-paced introduction/survey course. We will

• introduce many of the basic tasks in NLP and discuss why they are challenging

• present linguistic concepts and standard methods (maths/algorithms) often used to solve these tasks

• give you enough background to be able to read (some) current NLP research papers and take follow-on courses in sem 2

But we will not

• say much about cutting edge methods or heavy-duty machine learning (see ML courses and sem 2 NLP courses)
Relationship to other NLP courses

- Alternative NLP course: FNLP (10 pts, level 9/3rd yr, sem 2)
  - Covers some similar material, but not all
  - Less fast-paced, less open-ended assignments/exam.
  - But *does* assume Python programming and probability.
  - Good if you just want a taster.
  - You **may not** take both FNLP and ANLP.

- **ANLP is required** if you want to take sem 2 NLP courses:
  - Natural Language Understanding (recent advances, including neural network approaches).
  - Machine Translation (implement parts of MT system).
  - Topics in NLP (read and present papers).
Preparing for next week

• We will be starting with probabilistic models next week.

• If you haven’t taken a course on probability theory (or related), start working through the tutorial now (link on Readings section of website).

• Probabilistic material starts early to give you longer to absorb before the exam.

• Later parts of course will have more linguistics and little additional maths.
Labs start Wed/Thu this week!

Which lab to go to?

- Speech and Language Processing MSc: Thu 22 Sep, 16:10-17:30, Forrest Hill, Drill Hall

- All other students: **ignore what your timetable says.** Do this (also listed on course web page):
  - Birthday Jan-Apr: Thu 22 Sep, 16:10-17:30, FH, Drill Hall. (You can leave early if you need to go to Extreme Computing.)
  - Birthday Sep-Dec: Wed 21 Sep, 12:10-13:30, FH, 3.D02
Preparing for the first lab

- **Before your lab**: do the Preliminaries section of Lab 1. That is,
  - Get your DICE account and make sure you can log in to the lab machines in FH (or find a partner who can).
  - Read/work through the Introduction to DICE (linked from the lab) while at a DICE machine.