Empirical Methods in Natural Language Processing
Lecture 1
Introduction

(today’s slides based on those of Sharon Goldwater, Philipp Koehn, Alex Lascarides)

10 January 2018
What is Natural Language Processing?
What is Natural Language Processing?

Applications

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

Core technologies

- Language modelling
- Part-of-speech tagging
- Syntactic parsing
- Named-entity recognition
- Coreference resolution
- Word sense disambiguation
- Semantic Role Labelling
- ...

NLP lies at the intersection of computational linguistics and artificial intelligence. NLP is (to various degrees) informed by linguistics, but with practical/engineering rather than purely scientific aims. Processing speech (i.e., the acoustic signal) is separate.
This course

NLP is a big field! We focus mainly on core ideas and methods needed for technologies in the second column (and eventually for applications).

- Linguistic facts and issues
- Computational models and algorithms, especially using data ("empirical")
What are your goals?

Why are you here? Perhaps you want to:

- work at a company that uses NLP (perhaps as the sole language expert among engineers)

- use NLP tools for research in linguistics (or other domains where text data is important: social sciences, humanities, medicine, . . . )

- conduct research in NLP (or IR, MT, etc.)
What does an NLP system need to “know”?

- Language consists of many levels of structure
- Humans fluently integrate all of these in producing/understanding language
- Ideally, so would a computer!
Words

This is a simple sentence
This is a simple sentence.

be
3sg
present
Parts of Speech

This is a simple sentence

DT VBZ DT JJ NN

PART OF SPEECH

WORDS

MORPHOLOGY
This is a simple sentence

This is a simple sentence

Sharon Goldwater FNLP Lecture 1 11

Nathan Schneider ENLP (COSC/LING-572) Lecture 1 9
This is a simple sentence

be
3sg
present

SIMPLE1
having
few parts

SENTENCE1
string of words
satisfying the
grammatical rules
of a language

Semantics
Discourse

This is a simple sentence

But it is an instructive one.

Sharon Goldwater FNLP Lecture 1

Nathan Schneider ENLP (COSC/LING-572) Lecture 1
Why is NLP hard?

1. **Ambiguity** at many levels:

   - Word senses: *bank* (finance or river?)
   - Part of speech: *chair* (noun or verb?)
   - Syntactic structure: *I saw a man with a telescope*
   - Quantifier scope: *Every child loves some movie*
   - Multiple: *I saw her duck*

How can we model ambiguity, and choose the correct analysis in context?
Ambiguity

What can we do about ambiguity?

- non-probabilistic methods (FSMs for morphology, CKY parsers for syntax) return all possible analyses.

- probabilistic models (HMMs for POS tagging, PCFGs for syntax) and algorithms (Viterbi, probabilistic CKY) return the best possible analysis.

But the “best” analysis is only good if our probabilities are accurate. Where do they come from?
Statistical NLP

Like most other parts of AI, NLP is dominated by statistical methods.

• Typically more robust than earlier rule-based methods.

• Relevant statistics/probabilities are *learned from data*.

• Normally requires *lots of data* about any particular phenomenon.
Why is NLP hard?

2. **Sparse data** due to **Zipf’s Law**.

- To illustrate, let’s look at the frequencies of different words in a large text corpus.
- Assume “word” is a string of letters separated by spaces (a great oversimplification, we’ll return to this issue...)
Word Counts

Most frequent words in the English Europarl corpus (out of 24m word tokens)

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

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

But also, out of 93,638 distinct words (word types), 36,231 occur only once. Examples:

- cornflakes, mathematicians, fuzziness, jumbling
- pseudo-rapporteur, lobby-ridden, perfunctorily,
- Lycketoft, UNCITRAL, H-0695
- policyfor, Commissioneris, 145.95, 27a
Order words by frequency. What is the frequency of $n$th ranked word?
Plotting word frequencies

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

Word frequency vs. rank

Frequency vs. Rank
Rescaling the axes

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

Word frequency vs. rank, log 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? \( fr = k \) \( \Rightarrow \) \( f = \frac{k}{r} \) \( \Rightarrow \) \( \log f = \log k - \log r \)
Implications of Zipf’s Law

• Regardless of how large our corpus is, there will be a lot of infrequent (and zero-frequency!) words.

• In fact, the same holds for many other levels of linguistic structure (e.g., syntactic rules in a CFG).

• This means we need to find clever ways to estimate probabilities for things we have rarely or never seen.
Why is NLP hard?

3. **Variation**

- Suppose we train a part of speech tagger on the Wall Street Journal:

  Mr./NNP Vinken/NNP is/VBZ chairman/NN of/IN Elsevier/NNP N.V./NNP ,/ , the/DT Dutch/NNP publishing/VBG group/NN ./.
Why is NLP hard?

3. **Variation**

- Suppose we train a part of speech tagger on the Wall Street Journal:

  Mr./NNP Vinken/NNP is/VBZ chairman/NN of/IN Elsevier/NNP N.V./NNP ,/, the/DTD Dutch/NNP publishing/VBG group/NN ./.

- What will happen if we try to use this tagger for social media??

  ikr smh he asked fir yo last name

Twitter example due to Noah Smith
Why is NLP hard?

4. **Expressivity**

- Not only can one form have different meanings (ambiguity) but the same meaning can be expressed with different forms:

  She gave the book to Tom **vs.** She gave Tom the book
  Some kids popped by **vs.** A few children visited
  Is that window still open? **vs** Please close the window
Why is NLP hard?

5 and 6. **Context dependence** and **Unknown representation**

- Last example also shows that correct interpretation is context-dependent and often requires world knowledge.

- Very difficult to capture, since we don’t even know how to represent the knowledge a human has/needs: What is the “meaning” of a word or sentence? How to model context? Other general knowledge?
Organization of Topics (1/2)

Traditionally, NLP survey courses cover morphology, then syntax, then semantics and applications. This reflects the traditional form-focused orientation of the field, but this course will be organized differently, with the following units:

- **Introduction** (≈4 lectures): Getting everyone onto the same page with the fundamentals of text processing (Python 3/Unix) and linguistics.

- **N-grams** (≈2 lectures): Statistical modeling of words and word sequences.

- **Classification** (≈3 lectures): Approaches to classification that ignore linguistic structure within a sentence or document, focusing on the individual words/bags of words.

- **Sequential Prediction** (≈5 lectures): Techniques that assign additional linguistic information to words in sentences by modeling sequential relationships, including part-of-speech tagging and lexical semantic tagging.
Organization of Topics (2/2)

- **Hierarchical Sentence Structure** (≈5 lectures): Tree-based models of sentences that capture grammatical phrases and relationships (syntactic structure), as well as structured representations of within-sentence semantic relationships.

- **Other Learning Paradigms and Applications** (≈4 lectures): Models for characterizing words and text collections based on unlabeled data, or nonlinear models (neural networks) without hand-engineered features; and overviews of language technologies for text such as machine translation and question answering.
Backgrounds

This course has enrollment from multiple programs:

- Linguistics
- Computer Science
- possibly: Data Analytics; Communication, Culture & Technology; Math & Statistics

This means that there will be a diversity of backgrounds and skills, which is a fantastic opportunity for you to learn from fellow students. It also requires a bit of care to make sure the course is valuable for everyone.
What’s not in this course

- Formal language theory
- Computational morphology
- Logic-based compositional semantics
- Speech/signal processing, phonetics, phonology

(But see next 2 slides!)
Some Related Courses as of Spring 2018 (1/2)

In Linguistics:

- Intro to NLP (Amir Zeldes, last semester)
- Signal Processing (Corey Miller, last semester)
- **Statistical Machine Translation** (Achim Ruopp, this semester)
- Computational Corpus Linguistics (Zeldes, last semester)
- Computational Discourse Models (Zeldes, Fall 2016)
Some Related Courses as of Spring 2018 (2/2)

In Computer Science:

- **Statistical Machine Translation** (Achim Ruopp, this semester)
- Machine Learning (Mark Maloof, Spring 2017)
- Automated Reasoning (Maloof, last semester)
- Data Analytics (Lisa Singh, last semester)
- **Information Retrieval** (Nazli Goharian, this semester)
- Text Mining & Analysis (Goharian, last semester)
- **Web Search and Sense-making** (Yang, this semester)
Course organization

• Instructor: Nathan Schneider
• TAs: Sean Simpson, Austin Blodgett
• Lectures: MW 3:30–4:45, ICC 106
• Web site: for syllabus, schedule (lecture slides/readings/assignments): http://tiny.cc/enlp
  – Make sure to read the syllabus!
  – No hard-copy textbook; readings will be posted online.
• We will also use Canvas for communication once enrollment is finalized.