Introduction to Computational Linguistics: Introductory information

Sharon Goldwater

6 July 2015

What is Computational Linguistics?

- Using computers to address linguistic questions by analyzing linguistic data
  - Collecting attested forms of a construction from a corpus
  - Extracting phonetic measures from speech data
  - Performing complex statistical analyses
- Maybe...

What is Computational Linguistics?

- Implementing computational theories of language acquisition/processing/change
  - Simulating the spread of a linguistic change through a population
  - Predicting garden-path effects in sentence processing
  - Testing whether certain prosodic cues can help identify word boundaries
- Sure...

Comp Ling vs. Natural Language Processing

- Scientific goals
  - Data collection and analysis
  - Making predictions and testing theories (modelling!)
- Engineering goals
  - Building practical systems
  - Improving application-oriented performance measures

Core methods in CL and NLP

- Mathematical:
  - Probabilistic inference
  - Information theory/entropy
  - Networks/graphs
- Computational:
  - Probabilistic inference
  - Grammars and parsing algorithms
  - Finite-state machines

This course

- Provide grounding in many of these core methods
  - Mathematical and algorithmic issues
  - Example probabilistic models: n-gram models, HMMs, PCFGs
  - Example linguistic applications: phonology through semantics

Course structure

- Mondays: 2 hours lecture with break; Thursdays: 1 hour lecture, 1 hour lab
- labs posted online, can start ahead/work with others
- One assignment, due Thu 23 July
- grades: 20% lecture attendance, 30% lab participation, 50% assignment
- Schedule on web page:
  `http://homepages.inf.ed.ac.uk/sgwater/teaching/lsa2015/`

Prerequisites and preparation

- Must have previous experience in Python and basics of probability theory
- Please check 'software' section of web page and install appropriate Python/modules
- Textbook: *Speech and language processing, 2nd ed.*, by Jurafsky and Martin
- See web page:
  `http://homepages.inf.ed.ac.uk/sgwater/teaching/lsa2015/`

Auditing

- Will take auditors, subject to room capacity
- Those on waitlist have priority; email me if you are not yet on waitlist
- Auditors can access all course materials (inc labs) but shouldn't expect help during lab sessions
- See web page: http://homepages.inf.ed.ac.uk/sgwater/teaching/lsa2015/

Introduction to Computational Linguistics: Probability estimation

Sharon Goldwater

6 July 2015

School of informatics

A famous quote

> It must be recognized that the notion "probability of a sentence" is an entirely useless one, under any known interpretation of this term.

Noam Chomsky, 1969

A famous quote

> It must be recognized that the notion "probability of a sentence" is an entirely useless one, under any known interpretation of this term.

Noam Chomsky, 1969

- "useless": To everyone? To linguists?
- "known interpretation": What are possible interpretations?

Intuitive interpretation

- "Probability of a sentence" = how likely is it to occur in natural language
  - Consider only a specific language (English)
  - Not including meta-language (e.g. linguistic discussion)

P(She studies morphosyntax) > P(She studies more faux syntax)

Automatic speech recognition

Sentence probabilities (**language model**) help decide between similar-sounding options.

speech input

↓ (Acoustic model)

possible outputs

She studies morphosyntax
She studies more faux syntax
She's studies morph or syntax
...

↓ (Language model)

best-guess output

She studies morphosyntax

Machine translation

Sentence probabilities help decide word choice and word order.

non-English input

↓ (Translation model)

possible outputs

She is going home
She is going house
She is traveling to home
To home she is going
...

↓ (Language model)

best-guess output

She is going home

So, not "entirely useless", but...

- Sentence probabilities are clearly useful for language engineering.
- But what about linguistics?

Human sentence processing

Low probability sentences ⇒ processing difficulty

- As measured by reading speed, regressive eye movements, etc
- NB probabilities usually computed incrementally (word-by-word)
- Probabilistic models now commonplace in psycholinguistics

But, what about zero probability sentences?

the Archaeopteryx winged jaggedly amidst foliage
vs
jaggedly trees the on flew

- Neither has ever occurred before.
  ⇒ both have zero probability.
- But one is grammatical (and meaningful), the other not.
  ⇒ "Sentence probability" is useless to linguists interested in grammaticality (competence).

The logical flaw

- "Probability of a sentence" = how likely is it to occur in natural language.
- Sentence has never occurred before ⇒ sentence has zero probability ??
- More generally, is the following statement true?

Event has never occurred ⇒ event has zero probability

Events that have never occurred

- Each of these events has never occurred:

My hair turns blue
I injure myself in a skiing accident
I travel to Finland

- Yet, they clearly have differing (and non-zero!) probabilities.

Events that have never occurred

- Each of these events has never occurred:

My hair turns blue
I injure myself in a skiing accident
I travel to Finland

- Yet, they clearly have differing (and non-zero!) probabilities.
- Most sentences (and events) have never occurred.
  - This doesn't make their probabilities zero (or meaningless), but
  - it does make **estimating** their probabilities trickier.

Example: weather forecasting

What is the probability that it will rain tomorrow?

- To answer this question, we need
  - data: measurements of relevant info (e.g., humidity, wind speed/direction, temperature).
  - model: equations/procedures to estimate the probability using the data.
- In fact, to build the model, we will need data (including *outcomes*) from previous situations as well.

Example: weather forecasting

What is the probability that it will rain tomorrow?

- To answer this question, we need
  - data: measurements of relevant info (e.g., humidity, wind speed/direction, temperature).
  - model: equations/procedures to estimate the probability using the data.
- In fact, to build the model, we will need data (including *outcomes*) from previous situations as well.
- Note that we will never know the "true" probability of rain $P(\text{rain})$, only our estimated probability $\hat{P}(\text{rain})$.

Example: language model

What is the probability of sentence $\vec{w} = w_1 \dots w_n$?

- To answer this question, we need
  - data: words $w_1 \dots w_n$, plus a large corpus of sentences ("previous situations", or **training data**).
  - model: equations to estimate the probability using the data.
- Different models will yield different estimates, even with same data.
- Deep question: what model/estimation method do humans use?

How to get better probability estimates

Better estimates definitely help in language technology. How to improve them?

- **More training data.** Limited by time, money. (Varies a lot!)
- **Better model.** Limited by scientific and mathematical knowledge, computational resources
- **Better estimation method.** Limited by mathematical knowledge, computational resources

We will return to the question of how to know if estimates are "better".

Notation

- When the distinction is important, will use
  - $P(\vec{w})$ for *true* probabilities
  - $\hat{P}(\vec{w})$ for *estimated* probabilities
  - $P_{\mathrm{E}}(\vec{w})$ for estimated probabilities using a particular estimation method E.
- But since we almost always mean estimated probabilities, may get lazy later and use $P(\vec{w})$ for those too.

Example: estimation for coins

I flip a coin 10 times, getting 7T, 3H. What is $\hat{P}(\mathrm{T})$?

Example: estimation for coins

I flip a coin 10 times, getting 7T, 3H. What is $\hat{P}(\mathrm{T})$?

- **Model 1:** Coin is fair. Then, $\hat{P}(T) = 0.5$

Example: estimation for coins

I flip a coin 10 times, getting 7T, 3H. What is $\hat{P}(\mathrm{T})$?

- **Model 1:** Coin is fair. Then, $\hat{P}(T) = 0.5$
- **Model 2:** Coin is not fair. Then, $\hat{P}(T) = 0.7$ (why?)

Example: estimation for coins

I flip a coin 10 times, getting 7T, 3H. What is $\hat{P}(\mathrm{T})$?

- **Model 1:** Coin is fair. Then, $\hat{P}(T) = 0.5$
- **Model 2:** Coin is not fair. Then, $\hat{P}(T) = 0.7$ (why?)
- **Model 3:** Two coins, one fair and one not; choose one at random to flip 10 times. Then, $0.5 < \hat{P}(T) < 0.7$.

Example: estimation for coins

I flip a coin 10 times, getting 7T, 3H. What is $\hat{P}(\mathrm{T})$?

- **Model 1:** Coin is fair. Then, $\hat{P}(T) = 0.5$
- **Model 2:** Coin is not fair. Then, $\hat{P}(T) = 0.7$ (why?)
- **Model 3:** Two coins, one fair and one not; choose one at random to flip 10 times. Then, $0.5 < \hat{P}(T) < 0.7$.

Each is a **generative model**: a probabilistic process that describes how the data were generated.

Defining a model

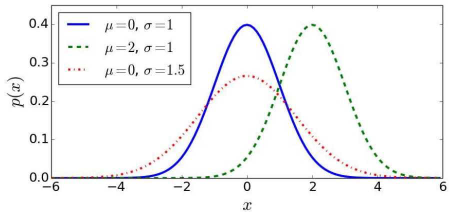
Usually, two choices in defining a model:

- **Structure** (or **form**) of the model: the form of the equations, usually determined by knowledge about the problem.
- **Parameters** of the model: specific values in the equations that are usually determined using the training data.

Example: height of 30-yr-old females

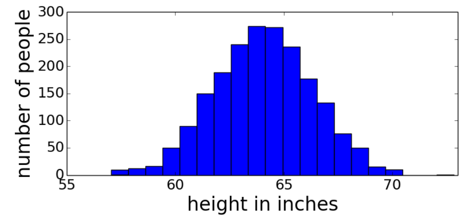
Assume the form of a **normal distribution**, with parameters (μ, σ):

$$p(x|\mu,\sigma) = \frac{1}{\sigma\sqrt{2\pi}} \exp\left(\frac{-(x-\mu)^2}{2\sigma^2}\right)$$



Example: height of 30-yr-old females

Collect data to determine values of μ, σ that fit this particular dataset.



Example: M&M colors

What is the proportion of each color of M&M?

- Assume a **discrete distribution** with parameters θ.
  - θ is a vector! That is, $\theta = (\theta_{\mathrm{R}}, \theta_{\mathrm{O}}, \theta_{\mathrm{Y}}, \theta_{\mathrm{G}}, \theta_{\mathrm{BL}}, \theta_{\mathrm{BR}})$.
  - For discrete distribution, params ARE the probabilities, e.g., $P(\text{red}) = \theta_{\mathrm{R}}$.

Example: M&M colors

What is the proportion of each color of M&M?

- Assume a **discrete distribution** with parameters θ.
  - θ is a vector! That is, $\theta = (\theta_{\mathrm{R}}, \theta_{\mathrm{O}}, \theta_{\mathrm{Y}}, \theta_{\mathrm{G}}, \theta_{\mathrm{BL}}, \theta_{\mathrm{BR}})$.
  - For discrete distribution, params ARE the probabilities, e.g., $P(\text{red}) = \theta_{\mathrm{R}}$.
- In 48 packages, I find[1] 2620 M&Ms, as follows:

Red	Orange	Yellow	Green	Blue	Brown
372	544	369	483	481	371

- How to estimate θ from this data?

[1] Actually I got the data from: https://joshmadison.com/2007/12/02/mms-color-distribution-analysis/

Relative frequency estimation

- Intuitive way to estimate discrete probabilities: **relative frequency** estimation.

$$P_{\mathrm{RF}}(x) = \frac{C(x)}{N}$$

where $C(x)$ is the count of x in a large dataset, and $N = \sum_{x'} C(x')$ is the total number of items in the dataset.

- M&M example: $P_{\mathrm{RF}}(\text{red}) = \hat{\theta}_{\mathrm{R}} = \frac{372}{2620} = .142$
- Or, could estimate probability of word w from a large corpus.
- Can we justify this mathematically?

Formalizing the estimation problem

- What is the best choice of θ given the data d that we saw?
- Formalize using Bayes' Rule, try to maximize $P(\theta|d)$.

$$P(\theta|d) = \frac{P(d|\theta)P(\theta)}{P(d)}$$

  - $P(\theta)$: **prior** probability of θ
  - $P(d|\theta)$: **likelihood**
  - $P(\theta|d)$: **posterior** probability of θ given d

Maximum-likelihood estimation

- Not obvious what prior should be: maybe just uniform?

$$\operatorname*{argmax}_{\theta} P(d|\theta)P(\theta) = \operatorname*{argmax}_{\theta} P(d|\theta)$$

- Choose θ to maximize the likelihood.
  - the parameters that make the observed data most probable
- This turns out to be just the relative frequency estimator, i.e.,

$$P_{\mathrm{ML}}(x) = P_{\mathrm{RF}}(x) = \frac{C(x)}{N}$$

Likelihood example

- For a fixed set of data, the likelihood depends on the model we choose.
- Our coin example, where $\theta = (\theta_{\mathrm{H}}, \theta_{\mathrm{T}})$. Suppose we saw $d = \text{HTTTHTHTTT}$.
- **Model 1:** Assume coin is fair, so $\hat{\theta} = (0.5, 0.5)$.
  - Likelihood of this model: $(0.5)^3 \cdot (0.5)^7 = 0.00097$

Likelihood example

- For a fixed set of data, the likelihood depends on the model we choose.
- Our coin example, where $\theta = (\theta_{\mathrm{H}}, \theta_{\mathrm{T}})$. Suppose we saw $d =$ HTTTHTHTTT.
- **Model 1:** Assume coin is fair, so $\hat{\theta} = (0.5, 0.5)$.
  - Likelihood of this model: $(0.5)^3 \cdot (0.5)^7 = 0.00097$
- **Model 2:** Use ML estimation, so $\hat{\theta} = (0.3, 0.7)$.
  - Likelihood of this model: $(0.3)^3 \cdot (0.7)^7 = 0.00222$
- Maximum-likelihood estimate does have higher likelihood!

Summary

- "Probability of a sentence": how likely is it to occur in natural language?
- Useful in natural language applications AND linguistics
- Can never know the true probability, but we may be able to estimate it.
- Probability estimates depend on
  - The data we have observed
  - The model (structure and parameters) we choose
- One way to estimate probabilities: maximum-likelihood estimation

Where to go from here?

Next time, we'll start to discuss

- Different generative models for sentences (model structure), and the questions they can address
- Weaknesses of MLE and ways to address them (parameter estimation methods)

First: one more piece of technical background.

Introduction to Computational Linguistics: Entropy

Sharon Goldwater

6 July 2015

School of informatics

Entropy

- Definition of **entropy**:

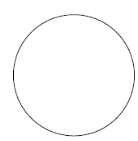
  $H(X) = \sum_x -p(x)\ \log_2 p(x)$
- Intuitively: a measure of uncertainty/disorder
- If we build a probabilistic model, we want that model to have low entropy (low uncertainty)

Entropy Example

One event (outcome)

$p(a) = 1$

$$\begin{aligned} H(X) &= -1 \log_2 1 \\ &= 0 \end{aligned}$$

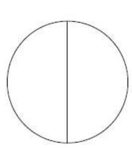


Entropy Example

2 equally likely events:

$p(a) = 0.5$
$p(b) = 0.5$

$$\begin{aligned} H(X) &= -0.5 \log_2 0.5 - 0.5 \log_2 0.5 \\ &= -\log_2 0.5 \\ &= 1 \end{aligned}$$

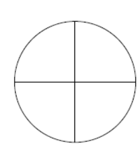


Entropy Example

4 equally likely events:

$p(a) = 0.25$
$p(b) = 0.25$
$p(c) = 0.25$
$p(d) = 0.25$

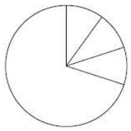
$$\begin{aligned} H(X) &= -0.25 \log_2 0.25 - 0.25 \log_2 0.25 \\ &\quad - 0.25 \log_2 0.25 - 0.25 \log_2 0.25 \\ &= -\log_2 0.25 \\ &= 2 \end{aligned}$$



Entropy Example

3 equally likely events and one more likely than the others:

$p(a) = 0.7$
$p(b) = 0.1$
$p(c) = 0.1$
$p(d) = 0.1$

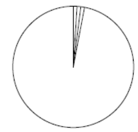


$$\begin{aligned} H(X) = & -0.7\log_2 0.7 - 0.1\log_2 0.1 \\ & -0.1\log_2 0.1 - 0.1\log_2 0.1 \\ = & -0.7\log_2 0.7 - 0.3\log_2 0.1 \\ = & -0.7 \times -0.5146 - 0.3 \times -3.3219 \\ = & \, 0.36020 + 0.99658 \\ = & \, 1.35678 \end{aligned}$$

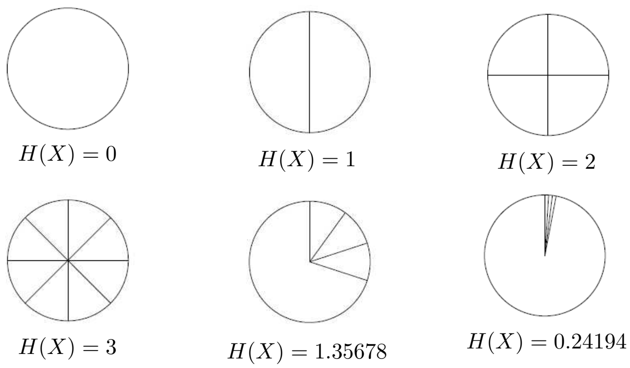
Entropy Example

3 equally likely events and one much more likely than the others:

$p(a) = 0.97$
$p(b) = 0.01$
$p(c) = 0.01$
$p(d) = 0.01$



$$\begin{aligned} H(X) = & -0.97\log_2 0.97 - 0.01\log_2 0.01 \\ & -0.01\log_2 0.01 - 0.01\log_2 0.01 \\ = & -0.97\log_2 0.97 - 0.03\log_2 0.01 \\ = & -0.97 \times -0.04394 - 0.03 \times -6.6439 \\ = & \, 0.04262 + 0.19932 \\ = & \, 0.24194 \end{aligned}$$



Entropy as y/n questions

How many yes-no questions (bits) do we need to find out the outcome?

- Uniform distribution with 2^n outcomes: n q's.
- Other cases: entropy is the average number of questions per outcome in a (very) long sequence, where questions can consider multiple outcomes at once.

Entropy as encoding sequences

- Assume that we want to encode a sequence of events X
- Each event is encoded by a sequence of bits
- For example
  - Coin flip: heads = 0, tails = 1
  - 4 equally likely events: a = 00, b = 01, c = 10, d = 11
  - 3 events, one more likely than others: a = 0, b = 10, c = 11
  - Morse code: *e* has shorter code than *q*
- Average number of bits needed to encode $X \geq$ entropy of X

The Entropy of English

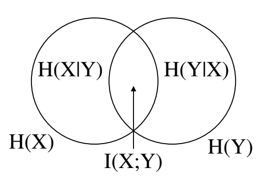
- Given a number of words in a text, can we guess the next word $p(w_n|w_1, ..., w_{n-1})$?
- Assuming a model with a limited window size (N = # words of history)

Model	Entropy
N=0	4.76
N=1	4.03
N=2	2.8
human, unlimited	1.3

Mutual Information

- A measure of independence between variables
  - How much (on average) does knowing Y reduce $H(X)$?

$$I(X;Y) = H(X) - H(X|Y)$$



  - Ex: on avg, how much more certain will I be about w_i if you tell me w_{i-1}?

Pointwise Mutual Information

- MI for two particular outcomes (no average)
- Definition:

$$I(x, y) = \log \frac{p(x, y)}{p(x)p(y)}$$

- Ex. Consider I(San, Francisco) vs. I(and, a)
- Will discuss more later in course