Topics in Natural Language Processing

Shay Cohen

Institute for Language, Cognition and Computation

University of Edinburgh

Lecture 2

Administrativia

Reminder: the requirements for the class are presentations, assignment, brief paper responses and an essay.

- Different topics are available online
- Example topics: topic models, language modeling, parsing, semantics, neural networks (your own topic?)
- Choose whatever level of difficulty you feel comfortable with, so that: (a) your presentation is clear; (b) your brief paper response is informative; (c) the essay goes into details about the topic.

Administrativia

- Presentations start on the week of 14/2
- Please submit the form I will send by Friday next week at 5pm (27/1)
- I will follow-up with an email by some time tomorrow

Today's Class

- Basic refresher about probability
- What is learning?
- What is a statistical model?
- How do we pick a statistical model?

Probability and Statistics: Reminder

Probability distribution? Example: unigram model

$\Omega = \{\text{the}, \text{cat}, \text{dog}, \text{sit}, \text{chase}\}$

$p\colon \Omega \to [0, 1]$ - $p(w)$ is the probability attached to w

$p(w) \geq 0$, $\sum_w p(w) = 1$, $\int_w p(w)dw = 1$

Random variables

Random variable:
A function $X\colon \Omega \to \mathbb{R}$

$\Omega = \{\text{the}, \text{dog}, \text{cat}\}$

$X_a(w) =$ count the number of a's in w

$X_a(\text{the}) = 0$, $X_a(\text{cat}) = 1$

$\Omega_2 = \{-\text{ed}, -\text{ing}, -\text{ion}\}$

$X(w) =$ suffix of the word, $X\colon \Omega \to \Omega_2$

Random variables induce probability distributions:

$p(X = \text{ion}) =$ the probability of a word w ending in -ion

$= \sum_{w:\ w \text{ ends in -ion}} p(w)$

$= \sum_{w} I(w \text{ ends in -ion}) p(w)$

$= E[I(w \text{ ends in ion})]$

where $I(\Gamma)$ is 0 if Γ is false and 1 if Γ is true.

Continuous random variables with density functions:
Guassians for example

Model Family

A set of probability distributions (unigram example):
$\mathcal{M} = \{p_1, p_2, \ldots\}$

$p_i \colon \Omega \rightarrow [0, 1]$

The model family does not have to be countable

Parameters

A set of parameters:
Θ where for each $\theta \in \Theta$ there is $p(w \mid \theta)$

$\mathcal{M} = \{p(w \mid \theta) \mid \theta \in \Theta\}$

$\Omega = \{\text{the}, \text{dog}, \ldots\}$

$p(w) =$ probability of word w
$\Theta \subset \mathbb{R}^{V-1}$ s.t. $0 \leq \theta_i \leq 1$

$\Theta \subset \mathbb{R}^{V}$ s.t. $0 \leq \theta_i \leq 1$ and $\sum_{i=1}^{V} \theta_i = 1$

Estimation

What is training data?

$w^{(1)}, w^{(2)}, w^{(3)}, \ldots \in \Omega$

Statistical Learning

- What does statistical learning do?
  - Induce a model from data
  - Models tell us how data are generated
  - Learning does the "opposite"

- Two different paradigms to Statistics: frequentist and Bayesian

Approach 1: frequentist Statistics

- We need an objective function $f(\theta, w_1, \ldots, w_n)$
- The higher the value of f is, the better it predicts the training data

$D = \{w_1, \ldots, w_n\}$

$D \rightarrow \Theta$ - that's estimation

$\theta^* = \arg\max_{\theta \in \Theta} f(\theta, w_1, \ldots, w_n)$

Choice of f: likelihood

$f(\theta, w_1, \ldots, w_n)$ is a real-valued function

$f(\theta, w_1, \ldots, w_n) = p(w_1, \ldots, w_n \mid \theta) = \prod_{i=1}^{n} p(w_i \theta)$

w_i are independent

Log-likelihood

$f(\theta, w_1, \ldots, w_n) = p(w_1, \ldots, w_n \mid \theta) = \prod_{i=1}^{n} p(w_i\theta)$

$\theta^* = \arg\max_\theta \prod_{i=1}^{n} p(w_i \mid \theta)$ – maximising likelihood

$L(w_1, \ldots, w_n) = \log f(\theta, w_1, \ldots, w_n)$

$\theta^* = \arg\max \log \left(\prod_{i=1}^{n} p(w_i \mid \theta)\right) = \arg\max_\theta \sum_{i=1}^{n} \log p(w_i \mid \theta)$

Next step

Estimation: maximisation of L. The result is the "best" θ that fits to the data *according to the objective function* L

$\theta^* = \arg\max_{\theta \in \Theta} \frac{1}{n} \sum_{i=1}^{n} \log p(w_i \mid \theta)$

The term maximised is called "average log-likelihood."

Pre-historic languages

Imagine a language with two words: “argh” and “blah”

Pre-historic languages

What is Ω?
$\Omega = \{\text{argh}, \text{blah}\}$

What is Θ?
$\Theta = [0, 1]$

θ is the probability of "argh"

$1 - \theta$ is the probability of "blah"

What is the training data?
$w^{(1)} = \text{argh}$, $w^{(2)} = \text{argh}$, $w^{(3)} = \text{blah}$, $w^{(1)} = \text{argh}$, ...

Pre-historic languages

What is the likelihood objective function?
$p(w_i \mid \theta) = \theta$ if $w_i = \text{argh}$ and $1 - \theta$ if $w_i = \text{blah}$.

$p(w_i \mid \theta) = \theta^{I(w_i = \text{argh})}(1-\theta)^{I(w_i = \text{blah})}$
What is the log-likelihood objective?
$\log p(w_i \mid \theta) = I(w_i = \text{argh}) \log \theta + I(w_i = \text{blah}) \log(1-\theta)$
$L(w_1, \ldots, w_n \mid \theta) = \sum_{i=1}^{n} \log p(w_i \mid \theta) = \sum_{i=1}^{n} I(w_i = a) \log \theta + (1 - I(w_i = b) \log(1-\theta)$

$$= \underbrace{\left(\sum_{i=1}^{n} I(w_i = a) \right)}_{a} \log \theta + \underbrace{\left(\sum_{i=1}^{n} 1 - I(w_i = b) \right)}_{b} \log(1-\theta)$$

$$= a \log \theta + b \log(1-\theta)$$

Pre-historic languages

Log-likelihood: $L(\theta, w_1, \ldots, w_n) = a \log \theta + b \log(1 - \theta)$

The maximisation problem: $\theta^* = \arg\max_\theta L(\theta, w_1, \ldots, w_n)$
$\frac{\partial L}{\partial \theta} = \frac{a}{\theta} - \frac{1}{1-\theta} \times b$
Equate derivative to 0
$a(1-\theta) - b\theta = 0$, note that $a + b = n$
Solution is
$\theta^* = \frac{a}{a+b} = \frac{a}{n}$

That's the maximum likelihood solution.

Principle of maximum likelihood estimation

- Objective function: log-likelihood (or likelihood)
- Estimation: maximise the log-likelihood with respect to the set of parameters

A guessing game

I choose a random number between 1 and 20. You need to guess it, and each time you make a guess I tell you whether your guess is higher or lower than my number. What is your strategy to guess the number as quickly as possible?

A guessing game

I choose a random number between 1 and 20. You need to guess it, and each time you make a guess I tell you whether your guess is higher or lower than my number. What is your strategy to guess the number as quickly as possible?

I choose a random number x between 1 and 20 **from a distribution** $p(x)$. You know p and need to guess the number. What is your strategy?

What does log-probability mean?

Let p be a probability distribution over Ω. What is $-\log_2 p(x)$?

Another view of maximum likelihood estimation

What is the “empirical distribution?”

Rewriting the objective function $L(\theta, w_1, \ldots, w_n)$

Cross-entropy

What is the definition of cross-entropy?

Likelihood maximisation

By doing maximum likelihood maximisation we:

- Choose the parameters that make the data most probable,

  or, from an information-theoretic perspective:

- Choose the parameters that make the encoding of the data most succinct (bit-wise),

  in other words, we

- Minimize the cross-entropy between the empirical distribution and the model we choose.

Types of Models

It is often the case that we discuss a model $p(x \mid \theta)$

Really, in NLP, you are interested in predicting some $y(x)$

Therefore, you need $p(x, y \mid \theta)$. Estimation is the same when both x and y are in the dataset. Later we will learn about incomplete data

In some cases you model also $p(y \mid x, \theta)$ (e.g. neural networks, log-linear models).

This gives the generative vs. discriminative model distinction

Types of Objectives

We showed an example of deriving the log-likelihood solution for a simple model

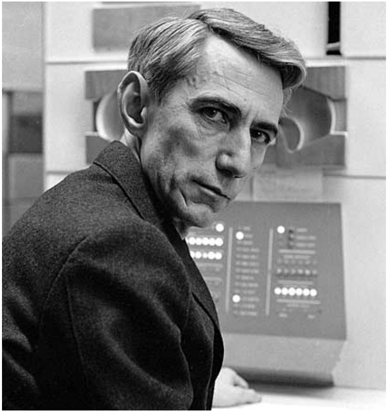
One can have more complex objective functions, and the principle would be the same

You just might not have a closed-form solution (e.g. with deep learning, log-linear models, etc.)

You need to apply an *optimization* algorithm – more on that later

A bit of history

One of the earliest experiments with statistical analysis of language – measuring entropy of English



2-3 bits are required for English

Approach 2: the Bayesian approach

- History: 1700s. Seminal ideas due to Thomas Bayes and Pierre-Simon Laplace

- A lot has changed since then...