# Topics in Natural Language Processing

Shay Cohen

Institute for Language, Cognition and Computation

University of Edinburgh

Lecture 1

# Topics in NLP

- We will cover the basic methodology in NLP
- There will be a focus on statistical learning
- Even more so, *structured prediction*

# Topics in NLP

**Prerequisites:**

- Some familiarity with machine learning and probability
- If something is unclear, ask!

**Things to Do:**

- Student presentations (20%)
- Brief paper responses (15%)
- Assignment (10%)
- Essay (55%)

**Office hours:** By appointment

# NLP in the Old Days

1950s-1980s: handwritten rules

IBM'S WATSON (right) AND FRIENDS:† For a mathematical wizard . . .

# NLP Now

late 1980s until now: statistical learning

# Learning

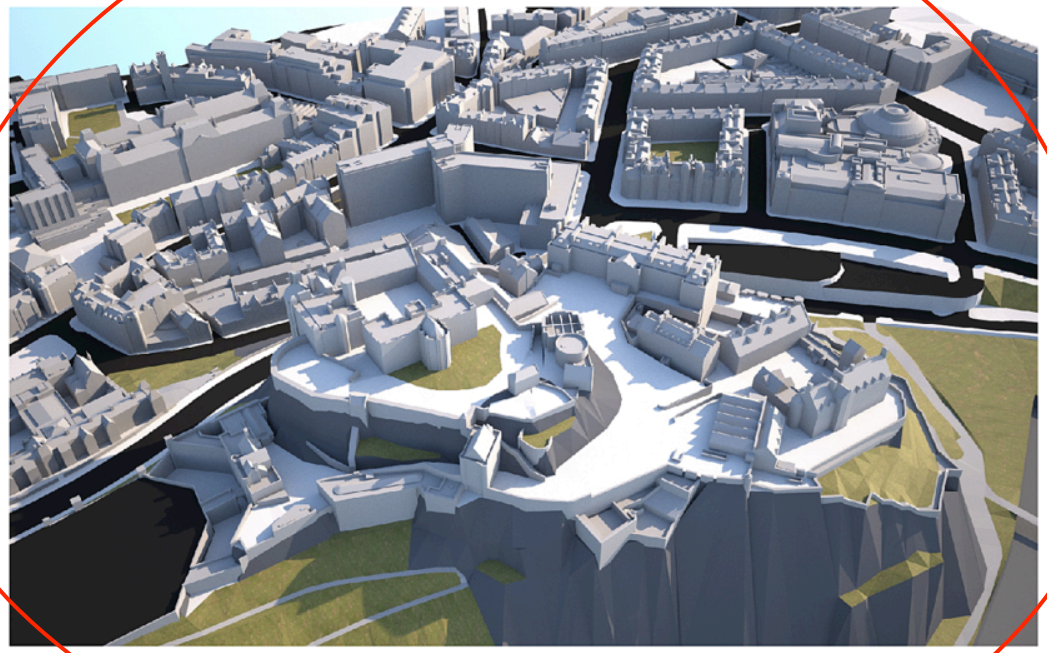
Learning is:

- Experience translated into expertise/knowledge
- Memorisation with generalisation

Machine learning and NLP:

- Experience = Training data
- Knowledge = Decoder or Prediction Model
- Used to either **mimic** humans or **transcend** their abilities

# What is a Model?



From Merriam-Webster:

- a usually small copy of something
- a set of ideas and numbers that describe the past, present, or future state of something (such as an economy or a business)

When is a model a good model?

# What is a Statistical Model?

Predict the future. Probabilistically.

# Probability and Statistics: Reminder

Probability distribution? Example: unigram model

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

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

$p(w) \geq 0$

$\sum_{w \in \Omega} p(w) = 1$

$\int_{w} p(w)\, dw = 1$

# Random variables

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

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

$X_a(\omega) =$ count of a's in $\omega$

$X_a(the) = 0 \qquad X_a(cat) = 1$

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

$X(\omega) =$ suffix of the word

$X : \Omega \to \Omega_2$

Random variables induce probability distribution:

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

$$= \sum_{\substack{w \\ \text{s.t. } w \text{ ends in ion}}} p(w)$$

$$= \sum_{w \in \Omega} I(\underbrace{w \text{ ends in -ion}}_{\Gamma}) \, p(w) =$$

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

$$\begin{cases} 1 & \text{if } \Gamma \text{ is true} \\ 0 & \text{o/w} \end{cases}$$

# Model Family

A set of probability distributions (unigram example):

model

model

$M = \{ p_1, p_2, \ldots \}$

$p_i : \Sigma \rightarrow [0, 1]$

# Parameters

A set of parameters: $\Theta$ for $\theta \in \Theta$ $p(w|\theta)$

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

$\Omega = \{$ the, dog, ... $\}$ $\quad |\Omega| = V$

$p(w)$ = prob. of word $w$

unigram

find a $\theta$ that characterizes what we'd expect from $p(w)$

$\Theta = \{ \theta \subseteq \mathbb{R}^{V-1}$ s.t. $1 \geq \theta_i \geq 0$ $\sum_{i=1}^{V} \theta_i = 1 \}$

# Another Parametrisation

Rely on properties of the words:

$\theta$ will be just a vector of length 26

$\theta_a \quad \theta_b \quad \theta_c, \ldots$

$$p(w) = \prod_{i=1}^{|w|} p(w_i)$$

# Estimation

What is training data?

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

# Estimation

What is the fit of the data to the model?

$f(w^{(1)}, \ldots w^{(n)}, \theta)$ - tells what is the fit of $\theta$ to $w^{(1)}, \ldots w^{(n)}$

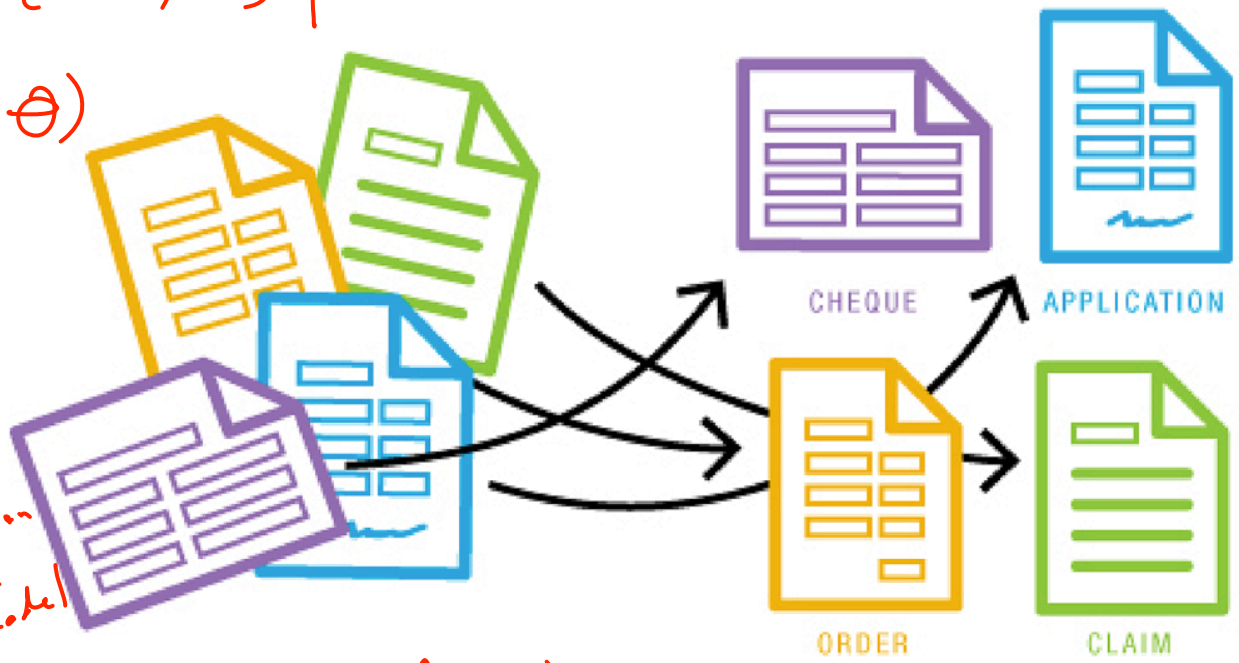
$$\theta^* = \arg\max_{\theta} f(w^{(1)}, \ldots w^{(n)}, \theta)$$

# NLP Problem Example: Document Classification

sentiment analysis, document topic, ...

$\Omega = \{(d, c) \mid c \text{ is a class, } d \text{ is a document}\}$

$p(d, c \mid \theta)$




prediction of the model

$\arg\max_c p(c \mid d, \theta)$

assignment

$\theta^* = \arg\max_\theta f((d^{(1)}, c^{(1)}), \ldots (d^{(n)}, c^{(n)}), \theta)$

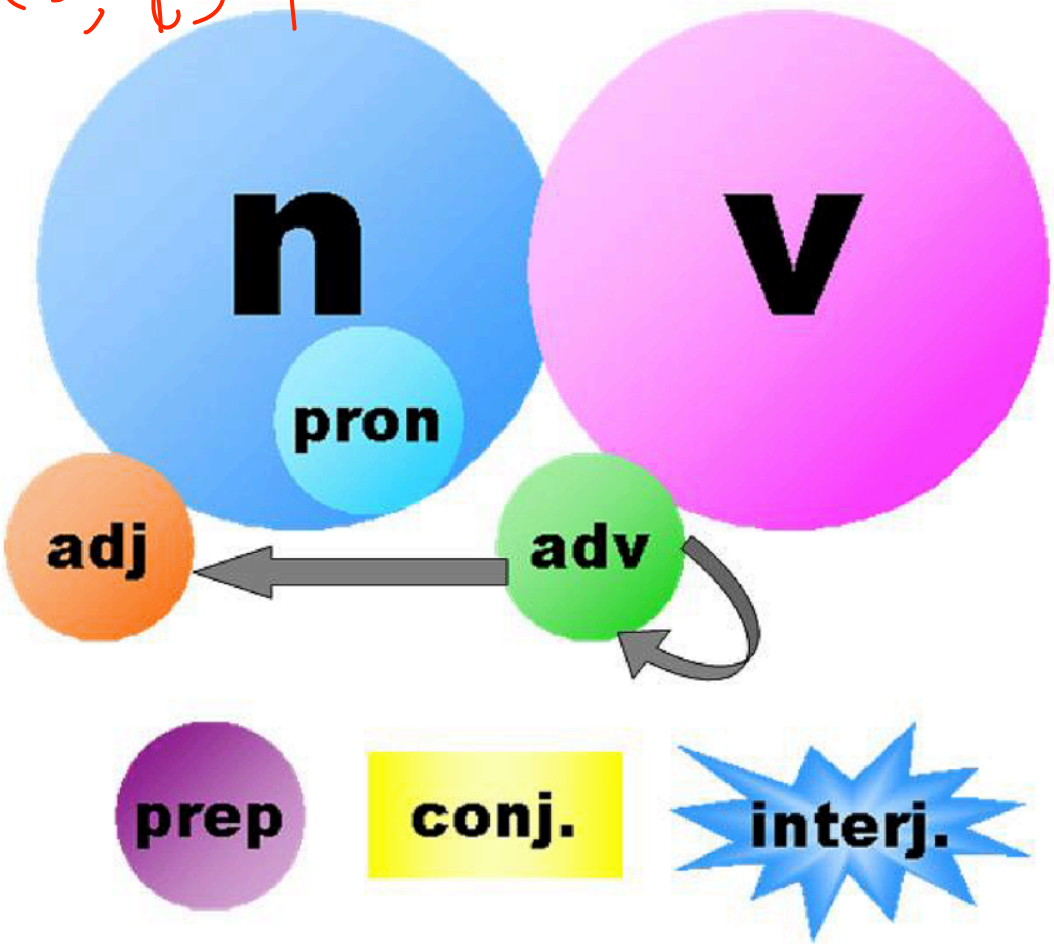
# NLP Problem Example: POS Tagging

map words to their part-of-speech tags

$\Omega = \{(s, t) \mid s \text{ is a sentence}, t \text{ is a tag}\}$

$p(s, t \mid \theta)$

$\arg\max_t p(t \mid s, \theta)$

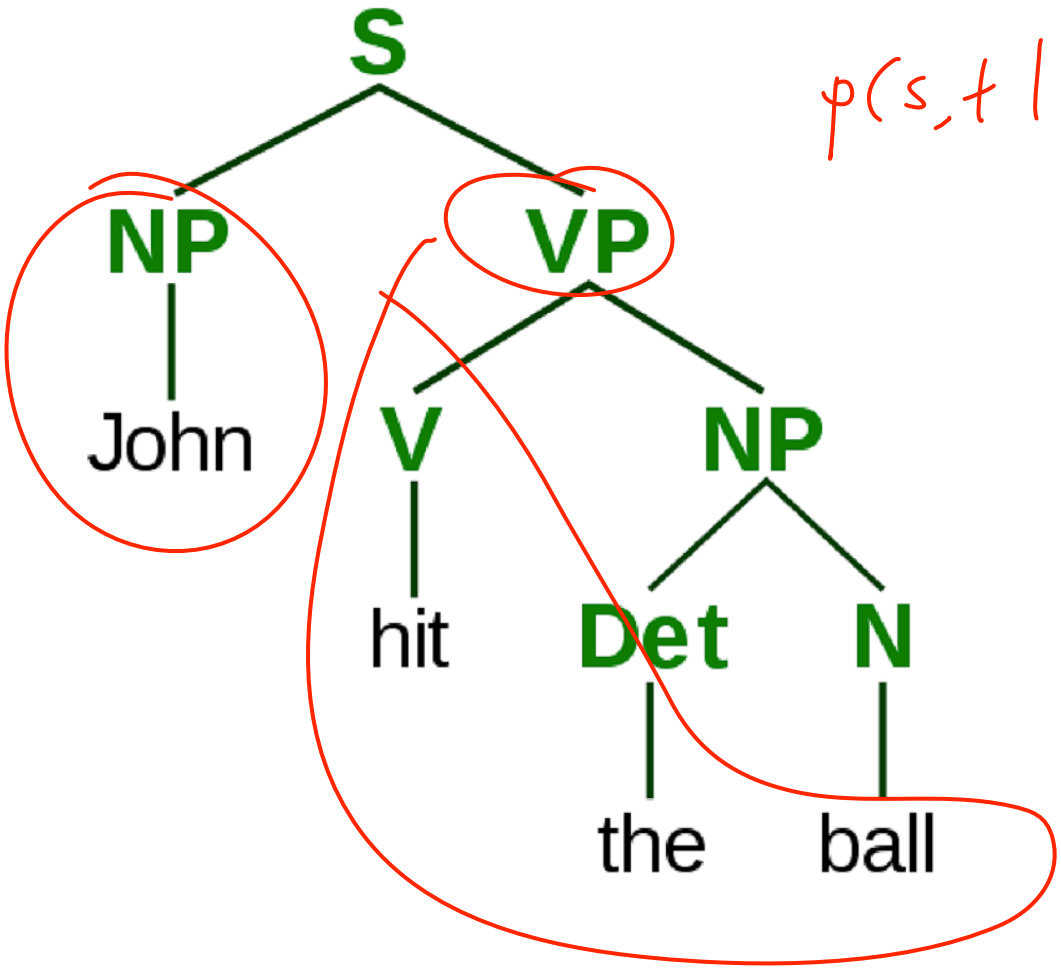


# NLP Problem Example: Parsing

map sentences to their syntax

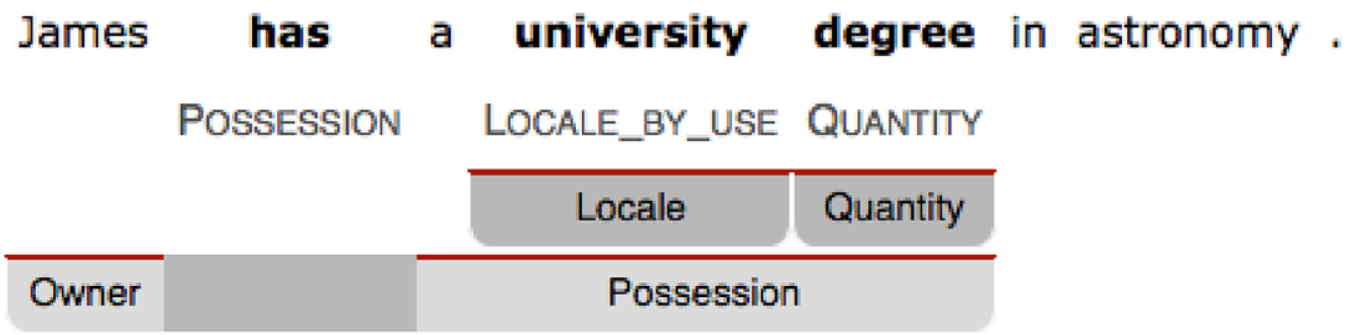
$\Omega = \{(s, t) \mid t \text{ is a tree}, s \text{ is sentence}\}$

$p(s, t \mid \theta)$



# NLP Problem Example: FrameNet Parsing

find predicate-argument structure



# Back to Modelling

What if the space to model is complex? Modelling documents.

# Modelling a Problem

- Define a sample space
- Define the structure of the sample space
- Decide on a parametrisation

Then one can proceed with data collection and learning

# Modelling - Tradeoffs

- “Exact copy”, detailed
- Not too many parameters
- Efficient to work with

# Next class

Paradigms in statistical learning

- Frequentist approaches
- Bayesian approaches
- “Computer science approaches?”