University of Oslo : Department of Informatics

INF4820: Algorithms for Artificial Intelligence and Natural Language Processing

Probabilities and Language Models

Stephan Oepen & Milen Kouylekov

Language Technology Group (LTG)

October 15, 2014

Introduction

So far: Point-wise classification (geometric models)

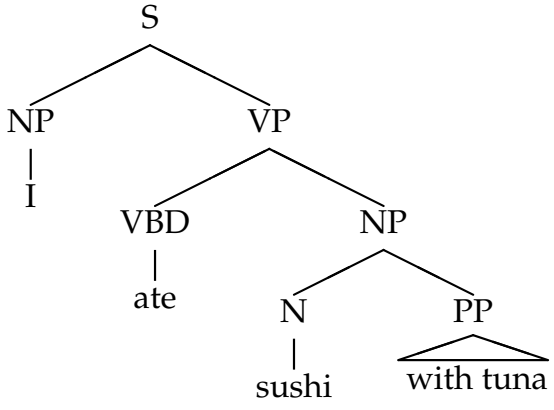
What's next: Structured classification (probabilistic models)

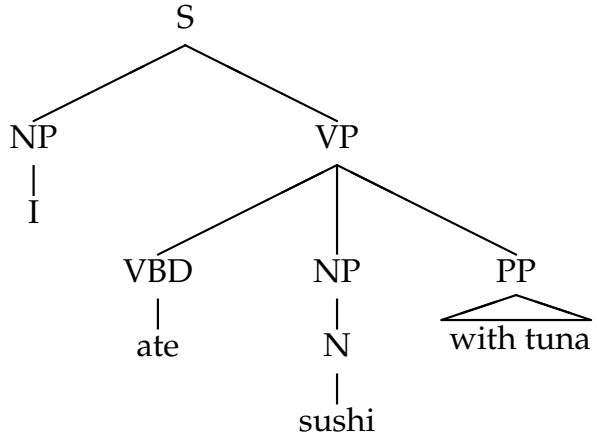
- sequences
- labelled sequences
- trees

By the End of the Semester . . .

. . . you should be able to determine

- which string is **most likely**:
  - *How to recognise speech* vs. *How to wreck a nice beach*
- which tag sequence is **most likely** for *flies like flowers*:
  - **NNS VB NNS** vs. **VBZ P NNS**
- which syntactic analysis is **most likely**:





Probability Basics (1/4)

- Experiment (or trial)
  - the process we are observing
- Sample space (Ω)
  - the set of all possible outcomes
- Events
  - the subsets of Ω we are interested in

$P(A)$ is the probability of event A, a real number $\in [0, 1]$

Probability Basics (2/4)

- Experiment (or trial)
  - rolling a die
- Sample space (Ω)
  - $\Omega = \{1, 2, 3, 4, 5, 6\}$
- Events
  - A = rolling a six: $\{6\}$
  - B = getting an even number: $\{2, 4, 6\}$

$P(A)$ is the probability of event A, a real number $\in [0, 1]$

Probability Basics (3/4)

- Experiment (or trial)
  - flipping two coins
- Sample space (Ω)
  - $\Omega = \{HH, HT, TH, TT\}$
- Events
  - A = the same both times: $\{HH, TT\}$
  - B = at least one head: $\{HH, HT, TH\}$

$P(A)$ is the probability of event A, a real number $\in [0, 1]$

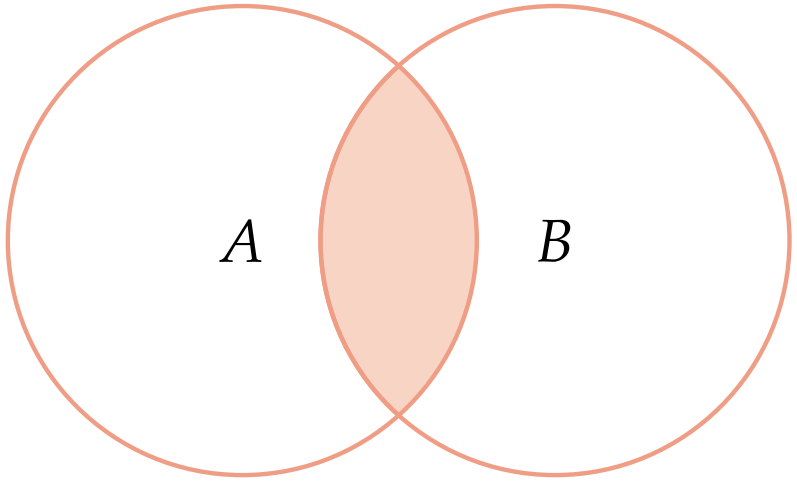
Probability Basics (4/4)

- Experiment (or trial)
  - rolling two dice
- Sample space (Ω)
  - $\Omega = \{11, 12, 13, 14, 15, 16, 21, 22, 23, 24, \ldots, 63, 64, 65, 66\}$
- Events
  - A = results sum to 6: $\{15, 24, 33, 42, 51\}$
  - B = both results are even: $\{22, 24, 26, 42, 44, 46, 62, 64, 66\}$

$P(A)$ is the probability of event A, a real number $\in [0, 1]$

Joint Probability

- $P(A, B)$: probability that both A *and* B happen
- also written: $P(A \cap B)$

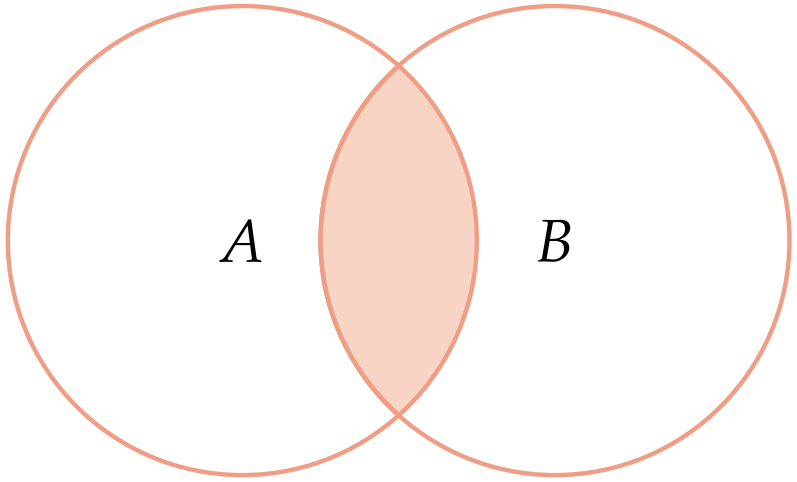


What is the probability, when throwing two fair dice, that

- A: the results sum to 6 *and*
- B: at least one result is a 1?

Joint Probability

- $P(A, B)$: probability that both A *and* B happen
- also written: $P(A \cap B)$

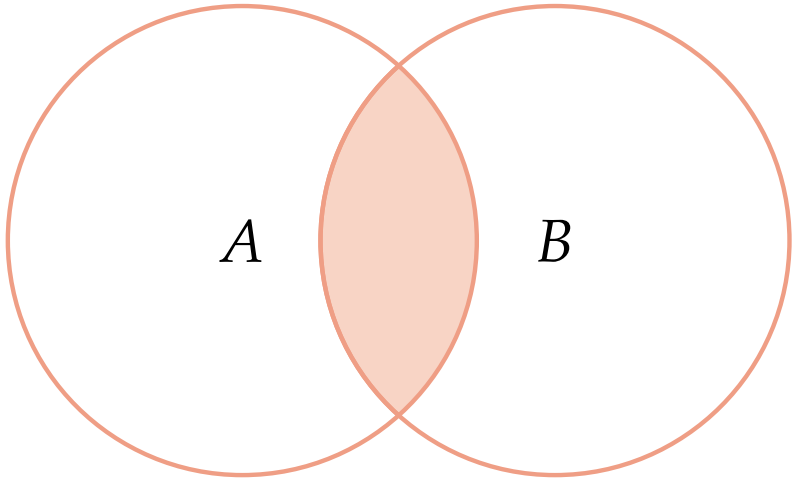


What is the probability, when throwing two fair dice, that

- A: the results sum to 6 *and*
- B: at least one result is a 1?

Joint Probability

- $P(A, B)$: probability that both A *and* B happen
- also written: $P(A \cap B)$

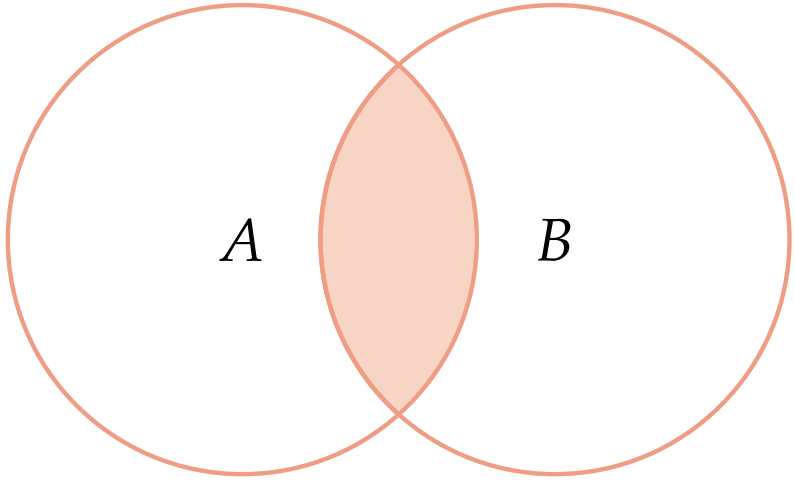


What is the probability, when throwing two fair dice, that

- A: the results sum to 6 *and* $\frac{5}{36}$
- B: at least one result is a 1?

Joint Probability

- $P(A, B)$: probability that both A *and* B happen
- also written: $P(A \cap B)$



What is the probability, when throwing two fair dice, that

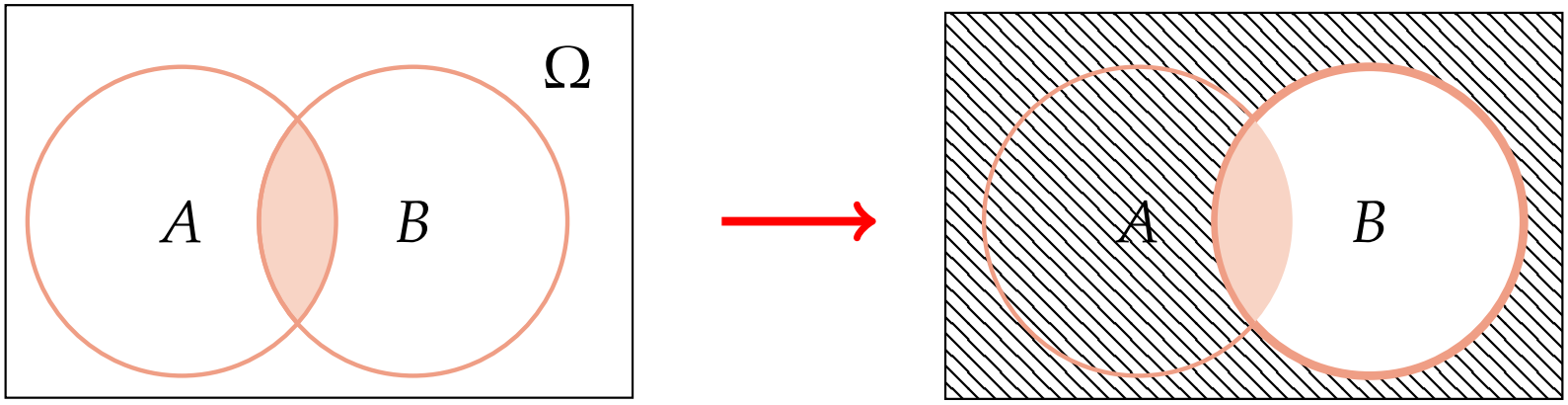
- A: the results sum to 6 *and* $\frac{5}{36}$
- B: at least one result is a 1? $\frac{11}{36}$

Conditional Probability

Often, we know *something* about a situation.

What is the probability $P(A|B)$, when throwing two fair dice, that

- A: the results sum to 6 *given*
- B: at least one result is a 1?



$$P(A|B) = \frac{P(A \cap B)}{P(B)} \quad \text{(where } P(B) > 0\text{)}$$

The Chain Rule

Since joint probability is symmetric:

$$\begin{aligned} P(A \cap B) &= P(A|B)\,P(B) \\ &= P(B|A)\,P(A) \quad \text{(multiplication rule)} \end{aligned}$$

More generally, using the *chain rule*:

$$P(A_1 \cap \cdots \cap A_n) = P(A_1)P(A_2|A_1)P(A_3|A_1 \cap A_2)\ldots P(A_n|\cap_{i=1}^{n-1} A_i)$$

The chain rule will be very useful to us through the semester:

- it allows us to break a complicated situation into parts;
- we can choose the breakdown that suits our problem.

(Conditional) Independence

If knowing event B is true has no effect on event A, we say

A and B are independent of each other.

If A and B are independent:

- $P(A) = P(A|B)$
- $P(B) = P(B|A)$
- $P(A \cap B) = P(A)\,P(B)$

Intuition? (1/3)

Let's say we have a rare disease, and a pretty accurate test for detecting it. Yoda has taken the test, and the result is positive.

The numbers:

- disease prevalence: 1 in 1000 people
- test false negative rate: 1%
- test false positive rate: 2%

What is the probability that he has the disease?

Intuition? (2/3)

Given:

- event A: have disease
- event B: positive test

We know:

- $P(A) = 0.001$
- $P(B|A) = 0.99$
- $P(B|\neg A) = 0.02$

We want

- $P(A|B) = ?$

Intuition? (3/3)

	A	¬ A	
B	**0.00099**	0.01998	**0.02097**
¬ B	0.00001	0.97902	0.97903
	0.001	0.999	1

$$P(A) = 0.001;\ P(B|A) = 0.99;\ P(B|\neg A) = 0.02$$

$$P(A \cap B) = P(B|A)P(A)$$

$$P(A|B) = \frac{P(A \cap B)}{P(B)} = \frac{0.00099}{0.02097} = 0.0472$$

Bayes' theorem

$$P(A|B) = \frac{P(B|A)P(A)}{P(B)}$$

- reverses the order of dependence
- in conjunction with the chain rule, allows us to determine the probabilities we want from the probabilities we have

Other useful axioms

- $P(\Omega) = 1$
- $P(A) = 1 - P(\neg A)$

Bonus: The Monty Hall Problem

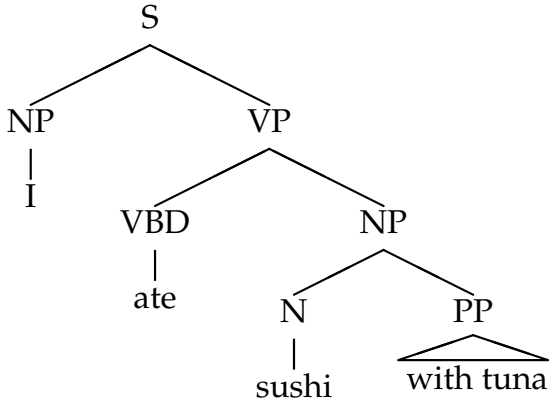
- On a gameshow, there are three doors.
- Behind 2 doors, there is a goat.
- Behind the 3rd door, there is a car.
- The contestant selects a door that he hopes has the car behind it.
- Before he opens that door, the gameshow host opens one of the other doors to reveal a goat.
- The contestant now has the choice of opening the door he originally chose, or switching to the other unopened door.

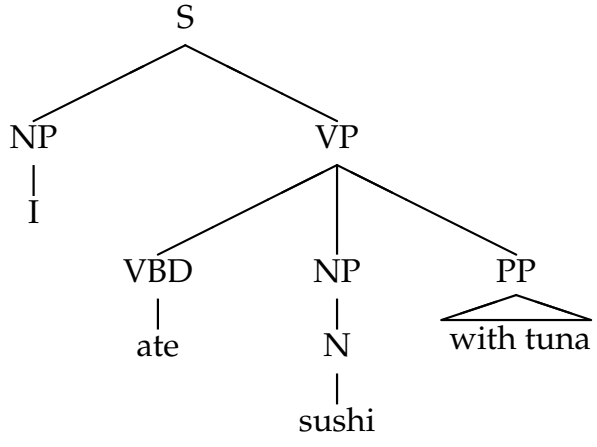
What should he do?

Recall Our Mid-Term Goals

Determining

- which string is most likely:
  - *How to recognise speech* vs. *How to wreck a nice beach*
- which tag sequence is most likely for *flies like flowers*:
  - **NNS VB NNS** vs. **VBZ P NNS**
- which syntactic analysis is most likely:





What Comes Next?

- Do you want to come to the movies and ___ ?
- Det var en ___ ?
- Je ne parle ___ ?

Natural language contains redundancy, hence can be predictable.

Previous context can constrain the next word

- semantically;
- syntactically;

→ by frequency.

Language Models

- A probabilistic (also known as stochastic) **language model** M assigns probabilities $P_M(x)$ to all strings x in language L.
  - L is the sample space
  - $0 \leq P_M(x) \leq 1$
  - $\sum_{x \in L} P_M(x) = 1$
- Language models are used in machine translation, speech recognition systems, spell checkers, input prediction, ...
- We can calculate the probability of a string using the chain rule:

$$P(w_1 \ldots w_n) = P(w_1)P(w_2|w_1)P(w_3|w_1 \cap w_2) \ldots P(w_n| \cap_{i=1}^{n-1} w_i)$$

$$P(\textit{I want to go to the beach}) =$$
$$P(\textit{I})\, P(\textit{want}|\textit{I})\, P(\textit{to}|\textit{I want})\, P(\textit{go}|\textit{I want to})\, P(\textit{to}|\textit{I want to go}) \ldots$$

N-Grams

We simplify using the **Markov assumption** (limited history):

the last $n-1$ elements can approximate the effect of the full sequence.

That is, instead of

- $P(beach|\ I\ want\ to\ go\ to\ the)$

selecting an n of 3, we use

- $P(beach|\ to\ the)$

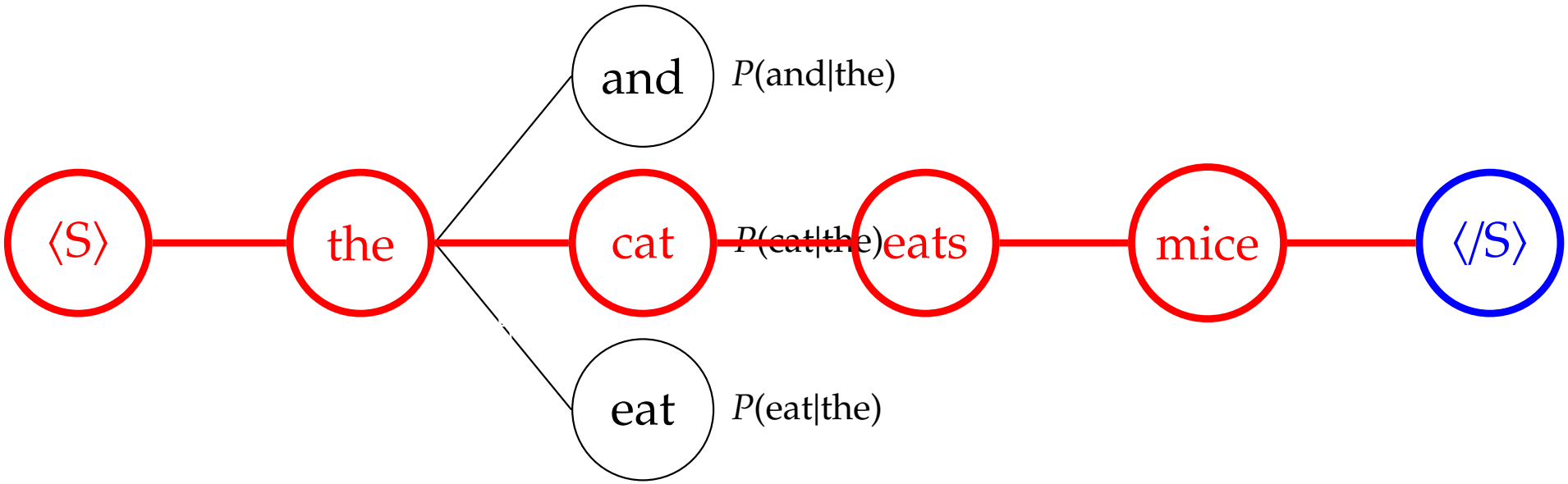
We call these short sequences of words n-grams:

- bigrams: *I want, want to, to go, go to, to the, the beach*
- trigrams: *I want to, want to go, to go to, go to the*
- 4-grams: *I want to go, want to go to, to go to the*

N-Gram Models

A generative model models a joint probability in terms of conditional probabilities.

We talk about the *generative story*:



$$P(S) = P(\text{the}|\langle S\rangle)\ P(\text{cat}|\text{the})\ P(\text{eats}|\text{cat})\ P(\text{mice}|\text{eats})\ P(\langle/S\rangle|\text{mice})$$

N-Gram Models

An n-gram language model records the n-gram conditional probabilities:

$$\begin{array}{llll} P(I|\langle S\rangle) & = 0.0429 & P(to|go) & = 0.1540 \\ P(want|I) & = 0.0111 & P(the|to) & = 0.1219 \\ P(to|want) & = 0.4810 & P(beach|the) & = 0.0006 \\ P(go|to) & = 0.0131 & & \end{array}$$

We calculate the probability of a sentence according to:

$$\begin{aligned} P\left(w_1^n\right) &\approx \prod_{k=i}^{n} P\left(w_k|w_{k-1}\right) \\ &\approx P\left(I|\langle S\rangle\right) \times P\left(want|I\right) \times P\left(to|want\right) \times P\left(go|to\right) \times P\left(to|go\right) \times \\ &\quad P\left(the|to\right) \times P\left(beach|the\right) \\ &\approx 0.0429 \times 0.0111 \times 0.4810 \times 0.0131 \times 0.1540 \times \\ &\quad 0.1219 \times 0.0006 = 3.38 \times 10^{-11} \end{aligned}$$

Training an N-Gram Model

How to estimate the probabilities of n-grams?

By counting (e.g. for trigrams):

$$P(\text{bananas}|\text{i like}) = \frac{C(\text{i like bananas})}{C(\text{i like})}$$

The probabilities are estimated using the **relative frequencies** of observed outcomes. This process is called **Maximum Likelihood Estimation** (MLE).

Bigram MLE Example

"I want to go to the beach"

w_1	w_2	$C(w_1 w_2)$	$C(w_1)$	$P(w_2 \mid w_1)$
$\langle S \rangle$	I	1039	24243	0.0429
I	want	46	4131	0.0111
want	to	101	210	0.4810
to	go	128	9778	0.0131
go	to	59	383	0.1540
to	the	1192	9778	0.1219
the	beach	14	22244	0.0006

What's the probability of *Others want to go to the beach* ?