03 Markov Models

Markov Property

Sequence: $\{x_1, x_2, x_3 \dots\}$ timeseries / words

Probability of sequence $p(x_1, x_2, x_3 \dots)$

Predict: $p(x_T | x_{T-1}, x_{T-2} \dots)$

Markov property: $$p(x_1 \dots x_T) = \underbrace{p(x_1)}_{\text{nothing before } x_1} \prod_{t=2}^{T} \underbrace{p(x_t | x_{t-1})}_{\text{only previous element influences}}$$

$$p(x_t | x_{t-1}, x_{t-2} \dots) = p(x_t | x_{t-1})$$

$\Rightarrow$ independent of $x_{t-2}, x_{t-3} \dots$

What if **MP not true?**

$$p(x_1, x_2) = p(x_1)\, p(x_2 | x_1)$$

$$p(x_1, x_2, x_3) = p(x_1) \underbrace{p(x_2 | x_1)\, p(x_3 | x_2, x_1)}_{p(x_2, x_3 | x_1)}$$

Chain rule of probability

$$p(x_1 \dots x_T) = p(x_1)\, p(x_2 | x_1)\, p(x_3 | x_1, x_2) \dots p(x_T | x_{T-1} \dots x_1)$$

Example

Find 10th word, given the first 9

$p(x_{10} | x_9 \dots x_1)$ x has 2000 possible words

Estimate 2000^{10} possibilities

Markov „Assumption"

eggs and ham   C++ and Python

word depends only on „and"

The Markov Model

$s(t) = s_t$ = state at time t

$t = 1, 2, 3, 4 \dots M$

$p(s_t = i)$ probab. state at $t = i$

State distribution

$p(s_t = 1)\ p(s_t = 2) \dots p(s_t = M)$

State transitions

$p(s_t = j \mid s_{t-1} = i)$

prob. state at t is j, given that state at $t-1$ was i

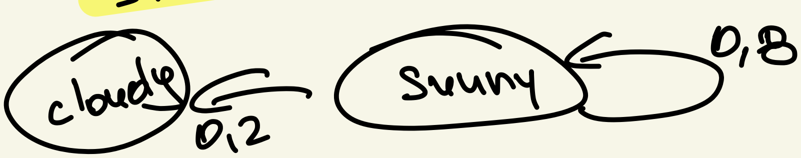
State transition Matrix

$$A_{ij} = p(s_t = j \mid s_{t-1} = i) \quad \forall i = 1 \dots M, j = 1 \dots M$$

first index = prev state
second index = next state

in general $A_{ij}(t)$ possible, but assume **time homogenious** Markov process: A_{ij} static

State transition Diagram



Initial State

initial state distribution

$\pi_i = p(s_1 = i)$ for $i = 1 \dots M$

π is $M \times 1$ $\quad A = M \times M$

Probability of sequence

$$p(s_{1 \dots T}) = p(s_1) \prod_{t=2}^{T} \underbrace{p(s_t \mid s_{t-1})}_{\text{state transition}}$$

$$p(s_{1 \dots T}) = \pi_{s_1} \prod_{t=2}^{T} A_{s_t s_{t-1}}$$

Training Markov Model

$$p(\text{heads}) \approx \frac{\text{count}(H)}{\text{total tosses}}$$

$$p(\text{"cat"}) \approx \frac{\text{count}(\text{cat})}{\text{total words}}$$

Estimating A / π

$$\hat{\pi}_i = \frac{\text{count}(s_1 = i)}{N}$$

→ seq. started with i

→ number of sequences

estimate with hat

$$\hat{A}_{ij} = \frac{\text{count}(i \to j)}{\text{count}(i)}$$

$$\frac{\text{count}(\text{the cat})}{\text{count}(\text{the})}$$

Probability smoothing / log prob.

A / π = max likelyhood estimate

$$P(s_1 \dots T) = \pi_{s_1} \prod_{t=2}^{T} A_{s_t s_{t-1}}$$

$I_k = 0$ since not appeared in training set (only in testing)

Add one smoothing

$$\hat{A}_{ij} = \frac{\text{count}(i \to j) + 1}{\text{count}(i) + M}$$

↳ each row of $A_{ij} = 1$

$$\hat{\pi}_i = \frac{\text{count}(s_1 = i) + 1}{N + M}$$

1 can be $\varepsilon \geq 1$ and $M = \varepsilon M$

more smooth $\varepsilon > 1$

less $\varepsilon < 1$

Compute prob sequence

A / π are very small (20-50k engl. words)

multiply gets closer to zero

compare 2 improbable sentences

Working with log probabilities

$A > B \quad \log(A) > \log(B)$

$\log_{10}(10^{-100}) = -100$

$$\log p(s_1 \dots T) = \log \pi_{s_1} + \sum_{t=2}^{T} \log A_{s_t s_{t-1}}$$

→ adding faster

Text Classifier

$\boxed{\text{Poem}} \to$ classify $\to$ poe / $\to$ Frost

email $\to$ spam

movie review $\to$ sentiment

Supervised or unsupervised

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

$\boxed{\text{Frost}} \to A_0\ \pi_0 \to p(x|\text{author}=\text{Frost})$

$\boxed{\text{PoE}} \to A_1\ \pi_1 \to p(x|\text{author}=\text{Poe})$

Apply Bayes' Rule

p(poem | author) but want p(author | poem)

$\Rightarrow k^* = \arg\max_k p(\text{class}=k \mid x)$

x = poem
k = author

$$p(\text{poem}|\text{auth}) = \frac{p(\text{poem}|\text{auth}) . p(\text{auth})}{p(\text{poem})}$$

$k^* = \arg\max_k p(\text{poem}|\text{author}=k)\, p(\text{author}=k)$

argmax (log.. + log..)

p(author) is uniform $\to$ const 50%

Max Likelyhood $k^* = \arg\max_k p(\text{poem}|\text{auth}=k)$

Markov Models to generate Text

Classifying text -> supervised
Generating -> unsupervised (no labels)

Discriminative
$p(y|x)$
no need to apply Bayes rule

Generative
$p(x|y)$
we find $p(y|x)$ later using Bayes rule

Sampling

$N(0,1)$ — 0: mean, 1: variance

Problems of Markov assumption

next word only depends on previous word

I made myself a peanut butter sandwich.
I will go and see her myself

Second order Markov Model:

$$p(s_t | s_{t-1}, s_{t-2} \dots) = p(s_t | s_{t-1}, s_{t-2})$$

2nd Order

$$A_{ijk} = p(s_t = k | s_{t-1} = j, s_{t-2} = i)$$

3D array (M^3 shape)
exp growth with number of past states

Full Model

$\pi_i = p(s_1 = i)$ for first word

$A^{(1)}_{ij}$ for second word

$A^{(2)}_{ijk}$ for every word after that

Article spinning

search engines detect duplicates

select a word to replace via a suggestion

N-Gram Approach

First order Markov Model for words

$$p(w_t \mid w_{t-1}) = \frac{count(w_{t-1} \rightarrow w_t)}{count(w_{t-1})}$$

$$p(w_t \mid w_{t-1}, w_{t-2}) = \frac{count(w_{t-2} \rightarrow w_{t-1} \rightarrow w_t)}{count(w_{t-2} \rightarrow w_{t-1})}$$

Predicting the middle word

$$p(w_t \mid w_{t-1}, w_{t+1}) = \frac{count(w_{t-1} \rightarrow w_t \rightarrow w_{t+1})}{count(w_{t-1} \rightarrow ANY \rightarrow w_{t+1})}$$

Max likelyhood est.

Production → Began → To

capacity

closer

continued

Facilities

noun/verb/adj

may not belong