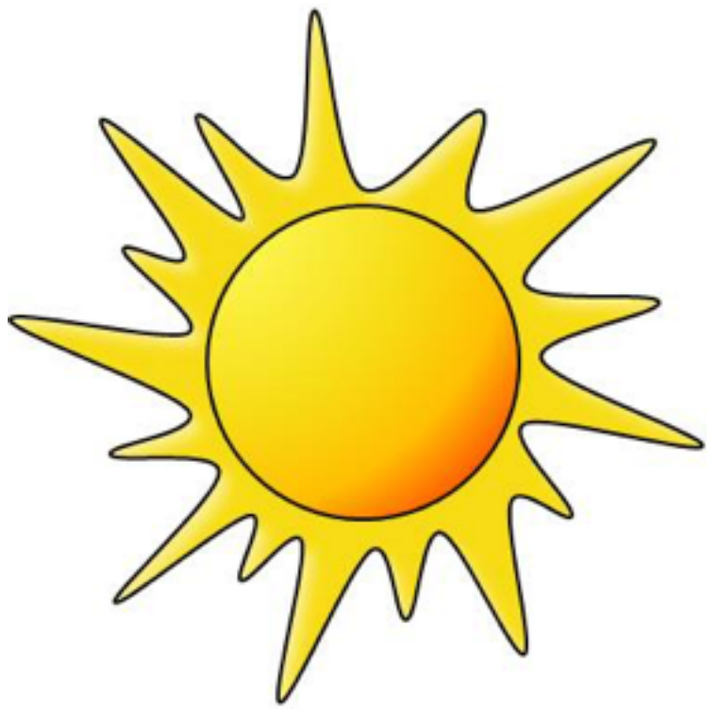
Language Modeling

Natural Language Processing

Emory University

Jinho D. Choi

Probability



Sunny

5 days

Cloudy

3 days

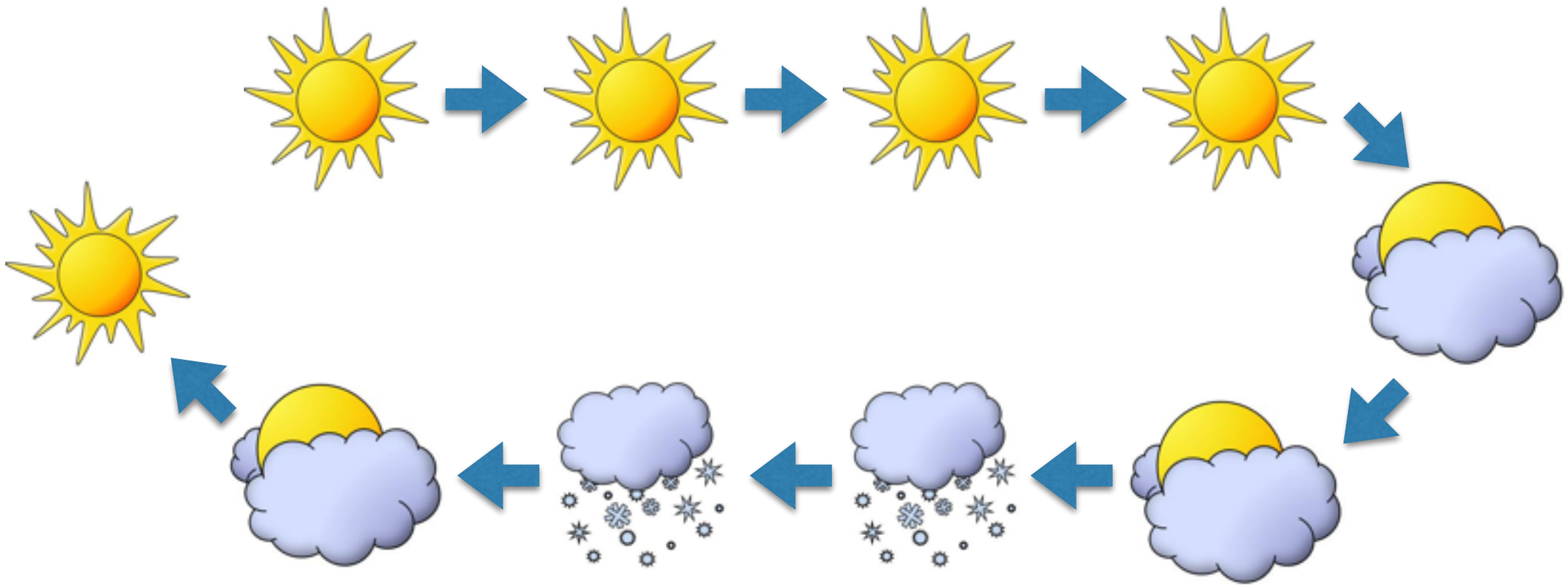
Snowy

2 days

Probability of tomorrow being cloudy?

$$P(cloudy) = \frac{C(cloudy)}{C(sunny) + C(cloudy) + C(snowy)} = \frac{3}{10}$$

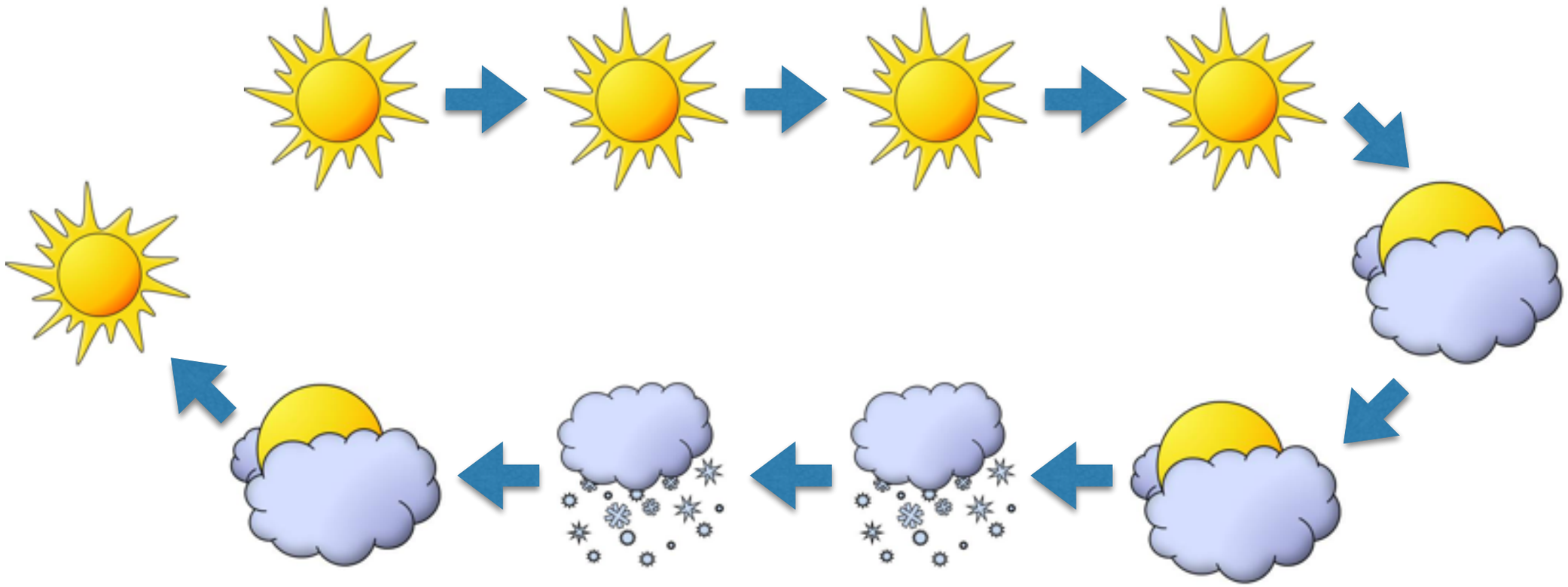
Conditional Probability



Probability of tomorrow being cloudy if today is snowy?

$$P(cloudy|snowy) = \frac{C(snowy, cloudy)}{C(snowy)} = \frac{1}{2}$$

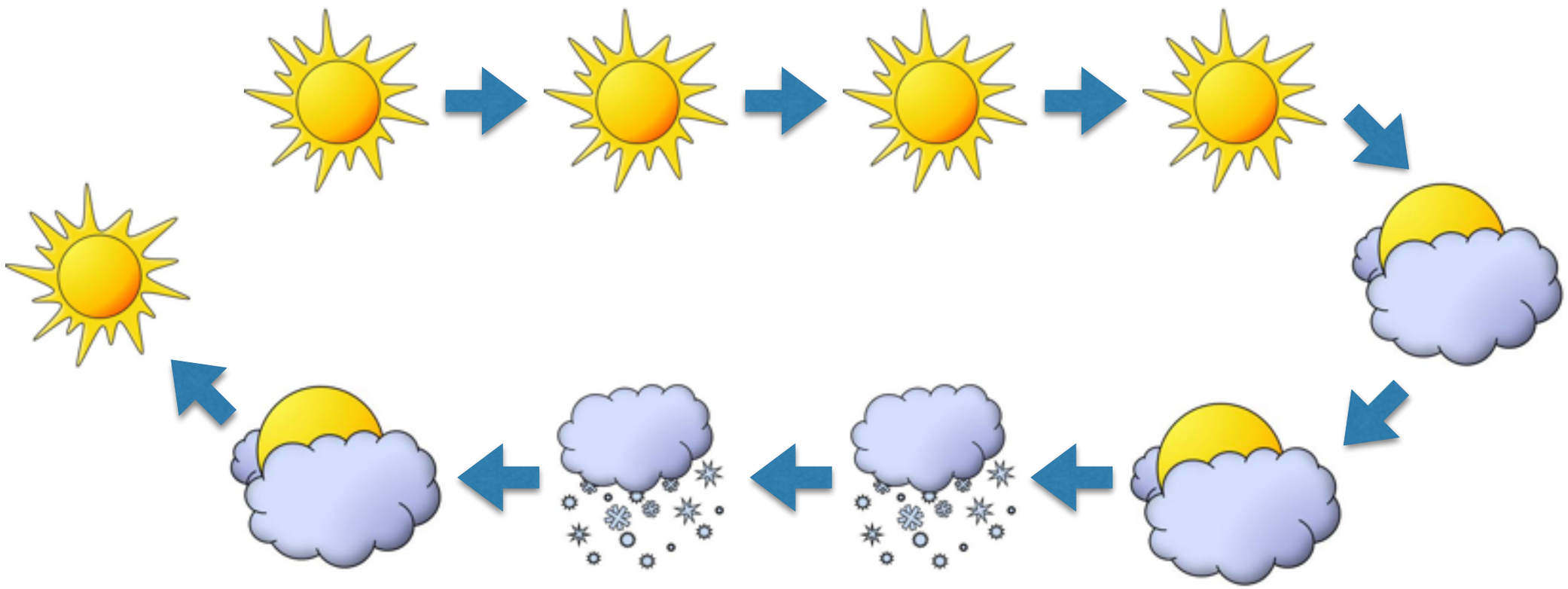
Conditional Probability



Probability of tomorrow being cloudy if today and yesterday are snowy?

$$P(cloudy|snowy, snowy) = \frac{C(snowy, snowy, cloudy)}{C(snowy, snowy)} = \frac{1}{1} = 1$$

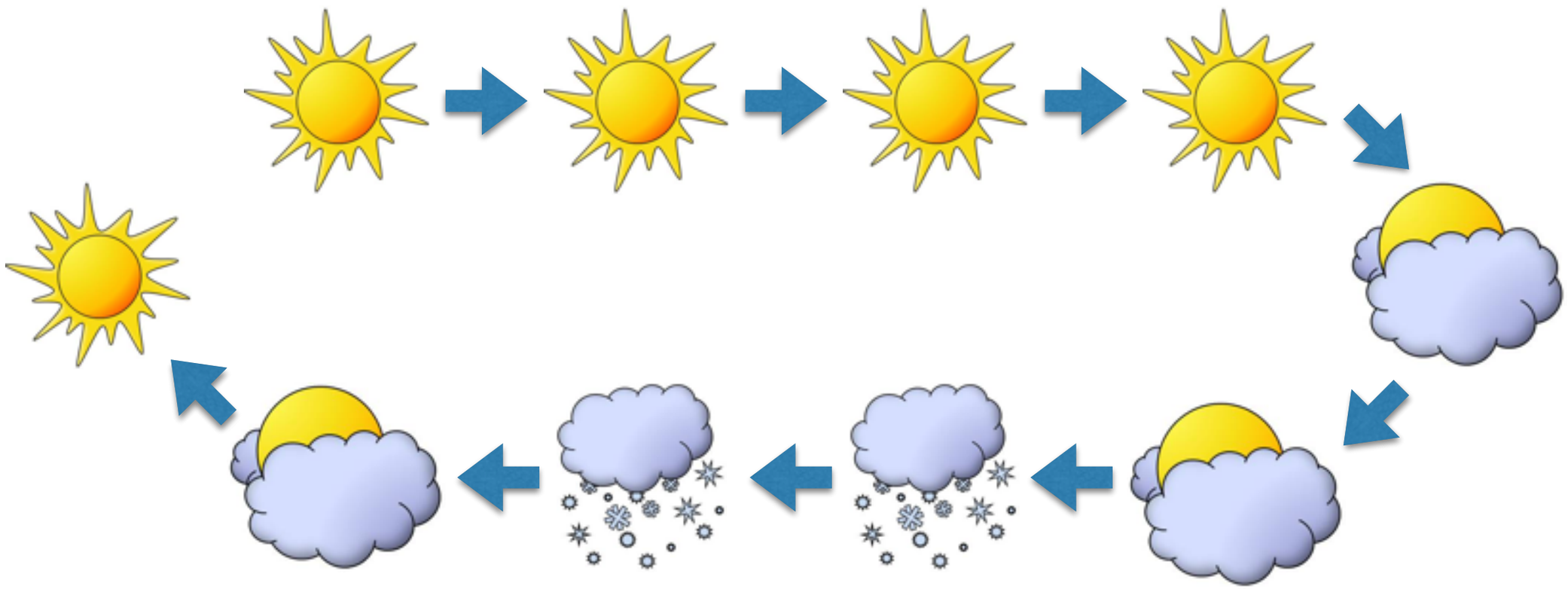
Joint Probability



Probability of 2 consecutive days being cloudy, sunny?

$$P(cloudy, sunny) = P(cloudy) \cdot P(sunny|cloudy)$$

Joint Probability



Probability of next 3 days being snowy, cloudy, sunny?

$$P(snowy, cloudy, sunny) = \begin{array}{l} P(snowy)\cdot \\ P(cloudy|snowy)\cdot \\ P(sunny|snowy, cloudy) \end{array}$$

N-gram Models

1-gram (Unigram)

$$P(w_i) = \frac{C(w_i)}{\sum_{\forall k} C(w_k)} = \frac{C(w_i)}{\boxed{N}}$$

# of tokens

token
vs
type?

2-gram (Bigram)

$$P(w_{i+1}|w_i) = \frac{C(w_i, w_{i+1})}{\sum_{\forall k} C(w_i, w_k)} = \frac{C(w_i, w_{i+1})}{C(w_i)}$$

N-gram Models

- Unigram model
  - Given any word w, it shows how likely w appears in context.
  - This is known as the likelihood (probability) of w, written as $P(w)$.
  - How likely does the word "Emory" appear in context?

Emory University was found as Emory College by John Emory .
Emory University is 16th among the colleges and universities in US .

$$P(Emory) = \frac{4}{23} \approx 0.1739$$

  - Does this mean "Emory" appears 17.39% time in any context?
  - How can we measure more accurate likelihoods?

N-gram Models

- Bigram model
  - Given any words w_i and w_j in sequence, it shows the likelihood of w_j following w_i in context.
  - This can be represented as the conditional probability of $P(w_j|w_i)$.
  - What is the most likely word following "Emory"?

Emory University was found as Emory College by John Emory.
Emory University is the 20th among the national universities in US.

$$\arg\max_k P(w_k|Emory)$$

$$\begin{aligned} P(University|Emory) &= \frac{2}{4} = 0.5 \\ P(College|Emory) &= \frac{1}{4} = 0.25 \\ P(.|Emory) &= \frac{1}{4} = 0.25 \end{aligned}$$

Maximum Likelihood

$$x_1^n = x_1, \ldots, x_n$$

Chain rule

$$P(x_1^n) = P(x_1) \cdot P(x_2|x_1) \cdot P(x_3|x_1^2) \cdots P(x_n|x_1^{n-1})$$

Any practical issue?

$(x_1, \ldots, x_k)$ can be very sparse.

Markov assumption

$$P(x_k|x_1^{k-1}) \approx P(x_k|x_{k-1})$$

$$P(x_1^n) \approx P(x_1) \cdot P(x_2|x_1) \cdot P(x_3|x_2) \cdots P(x_n|x_{n-1})$$

Maximum Likelihood

- Maximum likelihood
  - Given any word sequence $w_i, \ldots, w_n$, how likely this sequence appears in context.
  - This can be represented as the joint probability of $P(w_j, \ldots, w_n)$.
  - How likely does the sequence "you know" appears in context?

you know , I know you know that you do .

$$P(you, know) = \frac{2}{\boxed{11}}$$ not 10?

Chain rule

$$P(you) \cdot P(know|you) = \frac{3}{11} \cdot \frac{2}{3} = \frac{2}{11}$$

Maximum Likelihood

$$P(w_1, w_2, \ldots, w_n) = P(w_1^n)$$

$$P(w_1) \cdot P(w_2|w_1) \cdot P(w_3|w_1^2) \cdots P(w_n|w_1^{n-1})$$ Chain

$$\boxed{P(w_1)} \cdot P(w_2|w_1) \cdot P(w_3|w_2) \cdots P(w_n|w_{n-1})$$ Markov

$$P(w_1|\boxed{w_0}) \cdot P(w_2|w_1) \cdot P(w_3|w_2) \cdots P(w_n|w_{n-1})$$

Likelihood of a sequence of words

$$P(w_1^n) = \prod_{k=1}^{n} P(w_k|w_{k-1})$$

Log Likelihood

Benefit?

$$\boxed{\log} P(w_1^n) = \boxed{\log} \prod_{k=1}^{n} P(w_k|w_{k-1}) = \sum_{k=1}^{n} \boxed{\log} P(w_k|w_{k-1})$$

Word Segmentation

- Word segmentation
  - Segment a chunk of string into a sequence of words.
  - Are there more than one possible sequence?
  - Choose the sequence that most likely appears in context.

youknow

$$P(you) \cdot P(know|you) > P(yo) \cdot P(uk|yo) \cdot P(now|uk)$$

$$\log(P(you) \cdot P(know|you)) > \log(P(yo) \cdot P(uk|yo) \cdot P(now|uk))$$

$$\log(P(you)) + \log(P(know|you)) > \log(P(yo)) + \log(P(uk|yo)) + \log(P(now|uk))$$

Perplexity

How to evaluate a language model?

$$\mathrm{PL}(W) = P(w_1^n)^{-\frac{1}{n}} = \sqrt[n]{\frac{1}{P(w_1^n)}}$$

Inverse probability

Why?

$$\sqrt[n]{\prod_{k=1}^{n} \frac{1}{P(w_k|w_{k-1})}}$$

Branching factor?

Entropy

Entropy of word sequences whose length is *n*.

Measure of information

$$W_n = \{w_1, w_2, \ldots, w_n\}$$

$$H(W_n) = -\sum_{W'_n \in \mathcal{L}} p(W'_n) \log p(W'_n) \approx -\frac{1}{n} \log p(W'_n)$$

Entropy of language.

Shannon-McMillan-Breiman Theorem

$$H(\mathcal{L}) = \lim_{n\to\infty} \frac{1}{n} H(W_n)$$

$$= \lim_{n\to\infty} -\frac{1}{n} \sum_{W'_n \in \mathcal{L}} p(W'_n) \log p(W'_n)$$

$$= \lim_{n\to\infty} -\frac{1}{n} \log p(W'_n)$$

Entropy vs. Perplexity

$$\boxed{H(W_n)} = -\frac{1}{n}\log P(w_1^n)$$

$$\begin{aligned} 2^{H(W_n)} &= 2^{-\frac{1}{n}\log P(w_1^n)} \\ &= 2^{\log P(w_1^n)^{-\frac{1}{n}}} \\ &= P(w_1^n)^{-\frac{1}{n}} = \boxed{\mathrm{PL}(W_n)} \end{aligned}$$

Laplace Smoothing

$$P(x_1^n) \approx P(x_1) \cdot P(x_2|x_1) \cdot P(x_3|x_2) \cdots P(x_n|x_{n-1})$$

What if $P(x_1) = 0$? $\quad P(x_1^n) \approx 0 \quad \leftarrow$ BAD!!

$$P(x_i) = \frac{C(x_i)}{\sum_k C(x_k)}$$

Laplace Smoothing

$$P_l(x_i) = \frac{C(x_i) + \boxed{\alpha}}{\sum_k (C(x_k) + \boxed{\alpha})} = \frac{C(x_i) + \alpha}{\sum_k C(x_k) + \boxed{\alpha|X|}} = \frac{C(x_i) + \alpha}{N + \alpha|X|}$$

$$P_l(x_?) = \frac{\boxed{C(x_?) + \alpha}}{\sum_k C(x_k) + \alpha|X|} = \frac{\alpha}{N + \alpha|X|}$$

Laplace Smoothing

$$P(x_j|x_i) = \frac{C(x_i, x_j)}{\sum_k C(x_i, x_k)} = \frac{C(x_i, x_j)}{C(x_i)}$$

Laplace Smoothing

$$P_l(x_j|x_i) = \frac{C(x_i, x_j) + \alpha}{\sum_k (C(x_i, x_k) + \alpha)}$$

$$= \frac{C(x_i, x_j) + \alpha}{\sum_k C(x_i, x_k) + \boxed{\alpha|X_{i,*}|}}$$

$$= \frac{C(x_i, x_j) + \alpha}{C(x_i) + \alpha|X_{i,*}|}$$

$$P_l(x_?|x_i) = \frac{\boxed{\alpha}}{C(x_i) + \alpha|X_{i,*}|}$$

Discount Smoothing

- Issues with Laplace smoothing
  - Unfair discounts.

$$\frac{1}{100} = 0.01 \rightarrow \frac{1+1}{100+10} = 0.018 \quad +0.008$$

$$\frac{10}{100} = 0.1 \rightarrow \frac{10+1}{100+10} = 0.1 \quad 0$$

$$\frac{50}{100} = 0.5 \rightarrow \frac{50+1}{100+10} = 0.46 \quad -0.04$$

  - Unseen likelihood may get penalized too harshly when the minimum count is much greater than α.
  - How to reduce the gap between the minimum count and unseen count?

Discount Smoothing

Laplace	Discount
$P_l(x_?) = \frac{\alpha}{N + \alpha\lvert X\rvert}$	$P_d(x_?) = \alpha \cdot \min_k P(x_k)$
$P_l(x_i) = \frac{C(x_i) + \alpha}{N + \alpha\lvert X\rvert}$	$P_d(x_i) = \frac{C(x_i) - P_d(x_?)}{N}$
$P_l(x_?\lvert x_i) = \frac{\alpha}{C(x_i) + \alpha\lvert X_{i,*}\rvert}$	$P_d(x_?\lvert x_i) = \alpha \cdot \min_k P(x_k\lvert x_i)$
$P_l(x_j\lvert x_i) = \frac{C(x_i, x_j) + \alpha}{C(x_i) + \alpha\lvert X_{i,*}\rvert}$	$P_d(x_j\lvert x_i) = \frac{C(x_i, x_j) - P_d(x_?\lvert x_i)}{C(x_i)}$

Backoff

- Backoff
  - Bigrams are more accurate than unigrams.
  - Bigrams are sparser than unigrams.
  - Use bigrams in general, and use unigrams only bigrams don't exist.

$$P_b(x_j|x_i) = \begin{cases} P_{l|d}(x_j|x_i) & \boxed{P(x_j|x_i) > 0} \\ \boxed{\beta} \cdot P_{l|d}(x_j) & \text{Otherwise} \end{cases}$$

How to measure?

$$\beta = \alpha \cdot \boxed{\frac{\langle (P(x_j|x_i) \rangle_{i,j}}{\langle (P(x_j) \rangle_j}}$$

Interpolation

- Interpolation
  - Unigrams and bigrams provide different but useful information .
  - Use them both with different weights.

$$\hat{P}(x_j|x_i) = \lambda_1 \cdot P_{l|d}(x_j) + \lambda_2 \cdot P_{l|d}(x_j|x_i)$$

$$\sum_k \lambda_k = 1$$