Natural Language Processing (~1990)

1) Spam filtering

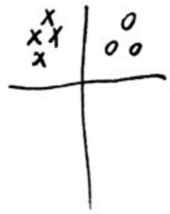
2) Query Completion

3) Document ranking

4) Topic extraction

Represent a text

1) Frequency counting



dimension = #words in your text

2) N-grams

n-gram is just a length n sequence from the text

- predict next word
- decrypting substitution Cyphers
- Filling in missing data

<u>Smoothing</u>: "The power of absolute discounting"

Hidden Markov Models

Markov Chain:

States : ~~$z_1, \dots, z_n$~~ $s_1, \cdots, s_n$

Transition probability $P = [P_{ij}]_{i,j=1}^{n}$

P_{ij} is the probability $S_i \to S_j$

z_t state of time t

$$\mathbb{P}(z_T = s \mid z_{T-1}, \cdots, z_1) = \mathbb{P}(z_T = s \mid z_{T-1})$$

$$\pi^t = [\tfrac{1}{3}, \tfrac{1}{3}, \tfrac{1}{3}]$$

$\pi \sim$ states, then $\pi_t = P^t \pi$

$\pi^* = \lim_{t \to \infty} \pi_t$ exists and is unique independent of π_0

π^* is limiting / equilibirium / stationary distribution

Hidden Markov Model:

States : $S_1, \cdots, S_n$

each state has "emission" distribution P_i

Have some variable $x_1, \cdots, x_k$

Goal: Estimate P ; P_i

EM algorithm

Markov Chain Monte Carlo (MCMC)

Idea: Model the distribution we want to estimate on the limiting distribution of MC. simmulate the Markov Chain.

Subtituition ~~Cybers~~ Cyphers

Σ Alphabet Goal: find σ^{-1}

$\sigma \in S_{|\Sigma|}$

1) $M =$ bigram conditional probability matrix $= [P_{ij}]$ for English

$\omega \in S_{|\Sigma|}$

$$pl(w) = \sum_{x=1}^{|T|-1} \log \cancel{w(T_k)} \left(P_{w(T_k) \cdot w(T_{k+1})} \right)$$

1) Pick ~~for~~ a random $w \in S_{|\Sigma|}$
2) loop until for $N \sim (2000)$
   3) Pick $\tau \in S_{|\Sigma|}$ a transposition
   4) If $pl(w.\tau) > pl(w)$

      $w = w \circ \tau$

      else with probability $\frac{pl(w \circ \tau)}{pl(w)}$

      $w = w \circ \tau$

Gibbs Sampling

$D_1, \dots, D_N$ documents

K topics

topic is a set of words and a distribution over the words

θ_i is a distribution on $T_1, \dots, T_k$

$(\theta_i)_j$ represents the porportion of D_i is T_j

1) Assign each word in each document a topic using θ_i's

2) For each document D, word $w \in D$

$$P(T|D) = \frac{\# \text{words in } D \text{ ~~arranged~~ assigned to } T}{\# \text{words in } D}$$

$$p(w|T) = \frac{\# \; w \text{ was assigned to } T}{\# \text{words in } T}$$

$$P(T|D) \cdot P(w|T)$$