
ADVANCED TOPICS IN INFORMATION RETRIEVAL AND WEB SEARCH

Lecture 5: Language Model-based IR

Dr: SM Vahidipour
vahidipour@kashanu.ac.ir

Based on the text book slides.

Outline

- **Language modeling**
- Language models for IR
- Smoothing
- Alternative models
- Comparison with traditional models

Language Model

- Probability distribution over strings of text
 - How likely is a given string (observation) in a given “language”?
 - Context-dependent!
 - Can also be regarded as a probabilistic mechanism for “generating” text, thus also called a “generative” model
 - Example:
 - $p1 = P(\text{“a quick brown dog”})$
 - $p2 = P(\text{“dog quick a brown”})$
 - $p3 = P(\text{“быстрая brown dog”})$
 - $p4 = P(\text{“быстрая собака”})$
- => in English: $p1 > p2 > p3 > p4$ (... depends on what “language” we are modeling)

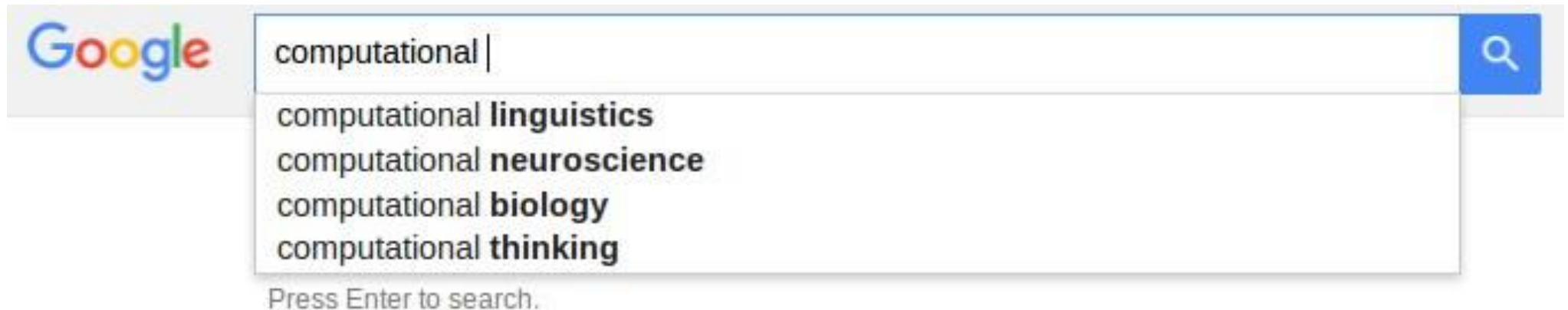
LM Usages

- Quantify the uncertainties in natural language
- Examples
 - Word prediction
 - Speech recognition
 - Text categorization
 - Information retrieval

LM Usages

- Example

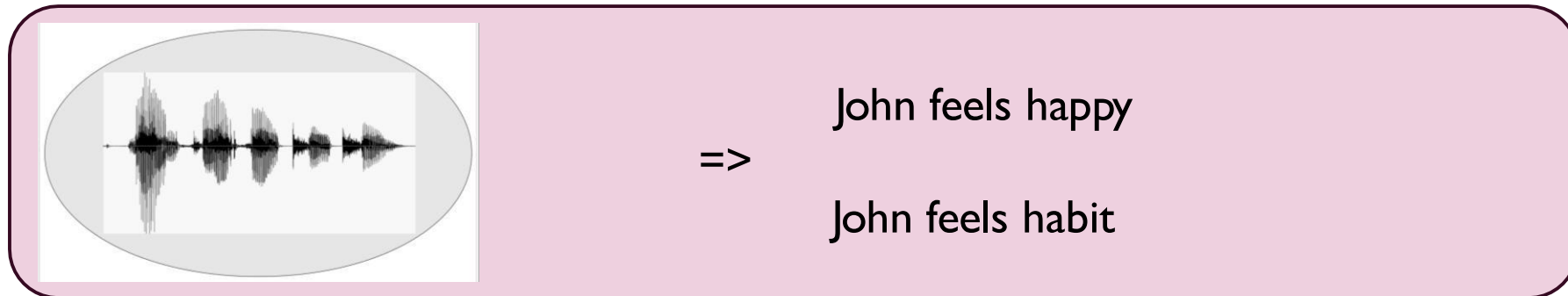
- Word prediction: given that a user types “computational”, how likely would he/she type “linguistics” as the next word compared to “biology”?



LM Usages

- Example

- Speech recognition: given that we see “John” and “feels”, how likely will we see “happy” as opposed to “habit” as the next word?



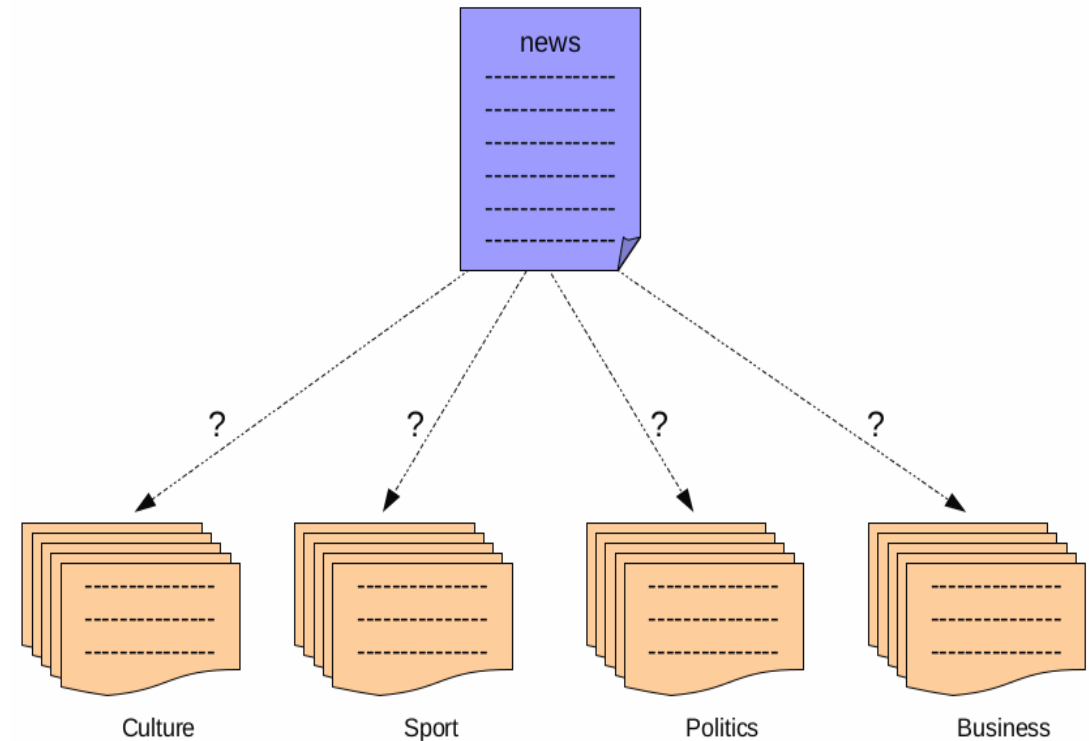
=>

John feels happy

John feels habit

LM Usages

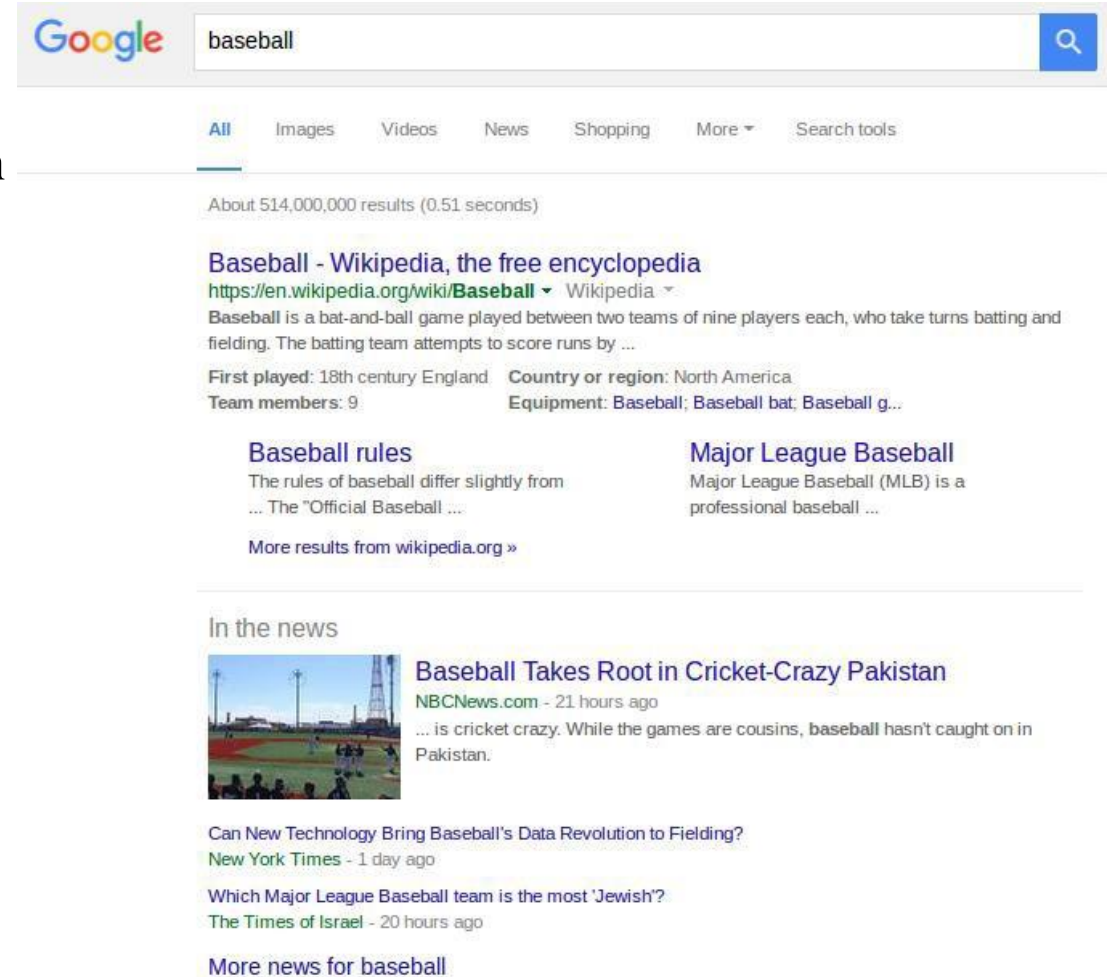
- Example
 - Text categorization: given that we observe “baseball” three times and “game” once in a news article, how likely is it about “sports”?



LM Usages

■ Example

- Information retrieval: given that a user is interested in sports news, how likely would the user use “baseball” in a query?



Google search for "baseball".

Search results include:

- Baseball - Wikipedia, the free encyclopedia**
<https://en.wikipedia.org/wiki/Baseball> - Wikipedia
Baseball is a bat-and-ball game played between two teams of nine players each, who take turns batting and fielding. The batting team attempts to score runs by ...
First played: 18th century England Country or region: North America
Team members: 9 Equipment: Baseball; Baseball bat; Baseball g...
- Baseball rules**
The rules of baseball differ slightly from ... The "Official Baseball ...
More results from wikipedia.org »
- Major League Baseball**
Major League Baseball (MLB) is a professional baseball ...

In the news

- Baseball Takes Root in Cricket-Crazy Pakistan**
NBCNews.com - 21 hours ago
... is cricket crazy. While the games are cousins, baseball hasn't caught on in Pakistan.
- Can New Technology Bring Baseball's Data Revolution to Fielding?**
New York Times - 1 day ago
- Which Major League Baseball team is the most 'Jewish'?**
The Times of Israel - 20 hours ago

More news for baseball

Outline

- Language modeling
- **Language models for IR**
- Smoothing
- Alternative models
- Comparison with traditional models

Main Papers

- The idea of using language model for IR was originally proposed in 1998

A Language Modeling Approach to Information Retrieval

Jay M. Ponte and W. Bruce Croft
Computer Science Department
University of Massachusetts, Amherst
{ponte, croft}@cs.umass.edu

Abstract Models of document indexing and document retrieval have been extensively studied. The integration of these two classes of models has been the goal of several researchers but it is a very difficult problem. We argue that much of the reason for this is the lack of an adequate indexing model. This suggests that perhaps a better indexing model would help solve the problem. However, we feel that making unwarranted parametric assumptions will not lead to better retrieval performance. Furthermore, making prior assumptions about the similarity of documents is not warranted either. Instead, we propose an approach to retrieval based on probabilistic language modeling. We estimate models

also to Harter [7]. By analogy to manual indexing, the task was to assign a subset of words contained in a document (the ‘specialty words’) as indexing terms. The probability model was intended to indicate the useful indexing terms by means of the differences in their rate of occurrence in documents ‘elite’ for a given term, i.e., a document that would satisfy a user posing that single term as a query, vs. those without the property of eliteness.

The success of the 2-Poisson model has been somewhat limited but it should be noted that Robertson’s *tf*, which has been quite successful, was intended to behave similarly to the 2-Poisson model [12].

Main Papers

- The important initial papers that originated the language modeling approach to IR are:
 - Ponte, Jay M., and W. Bruce Croft. 1998. A language modeling approach to information retrieval. In Proc. SIGIR, pp. 275–281. ACM Press.
 - Hiemstra, Djoerd. 1998. A linguistically motivated probabilistic model of information retrieval. In Proc. ECDL, volume 1513 of LNCS, pp. 569–584.
 - Berger, Adam, and John Lafferty. 1999. Information retrieval as statistical translation. In Proc. SIGIR, pp. 222–229. ACM Press.
 - Miller, David R. H., Tim Leek, and Richard M. Schwartz. 1999. A hidden Markov model information retrieval system. In Proc. SIGIR, pp. 214–221. ACM Press.
 - Lafferty, John, and Chengxiang Zhai. 2001. Document language models, query models, and risk minimization for information retrieval. In Proc. SIGIR, pp. 111–119. ACM Press.
 - Chengxiang Zhai. And Lafferty, John, 2004. A study of smoothing methods for language models applied to information retrieval, *ACM Transactions on Information Systems*, Vol. 2, Issue 2.

Using Language Models in IR

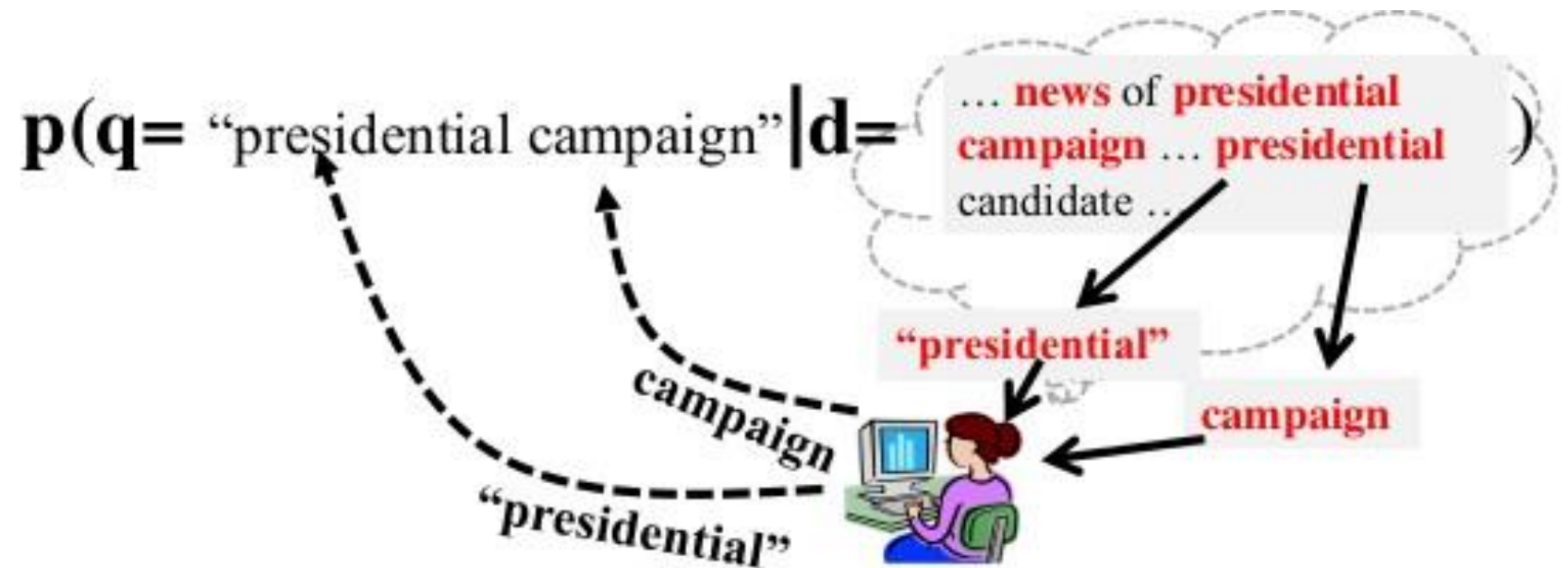
- Given a query Q
- Rank documents based on $P(D|Q)$

$$P(D|Q) = \frac{P(Q|D)P(D)}{P(Q)}$$

- $P(Q)$ is the same for all documents, so ignore
- $P(D)$ is the prior – often treated as the same for all D
 - But we can give a higher prior to “high-quality” documents, e.g., those with high PageRank.
- $P(Q|D)$ is the probability of Q given $D \Rightarrow$ query likelihood

Language Model for IR

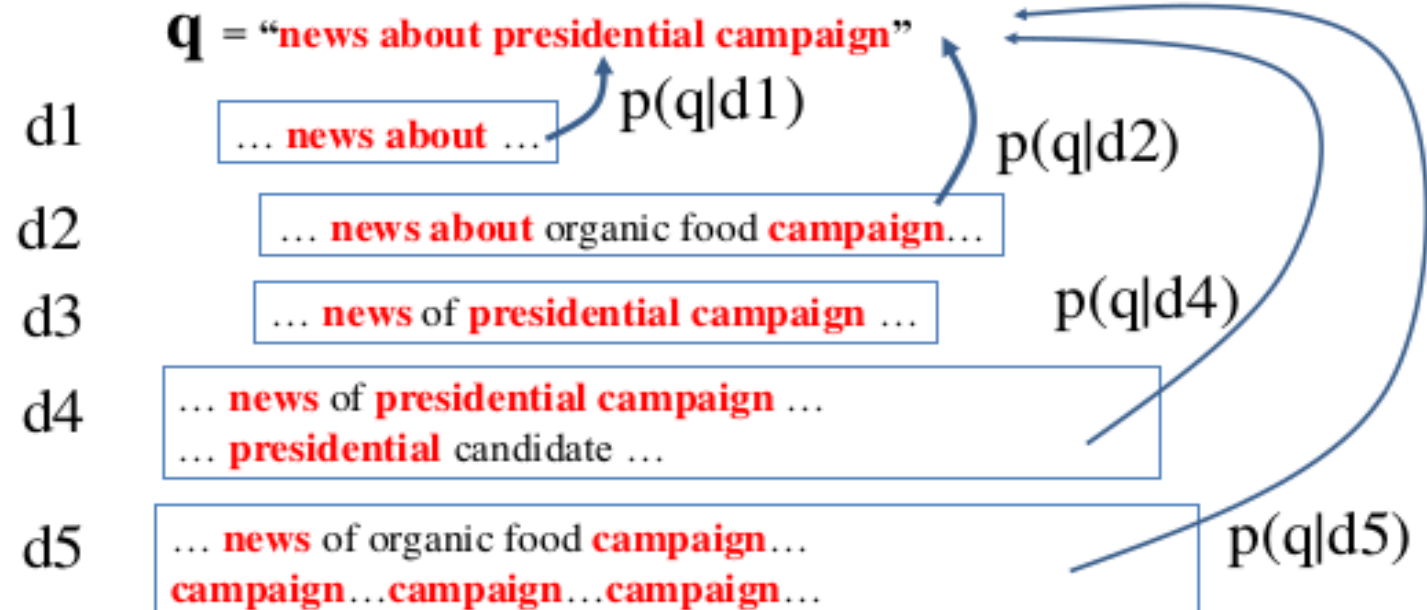
- Assumption: a user formulates a query based on an “imaginary relevant document”
- If the user is thinking of this doc, how likely would he/she pose this query?
- Query generation by sampling words from doc



Language Model for IR

- Which doc is Most Likely the “Imaginary Relevant Doc”?

$$p(q = \text{“presidential campaign”} | d = \text{... news of presidential campaign ... presidential candidate ...})$$



Language Model for IR

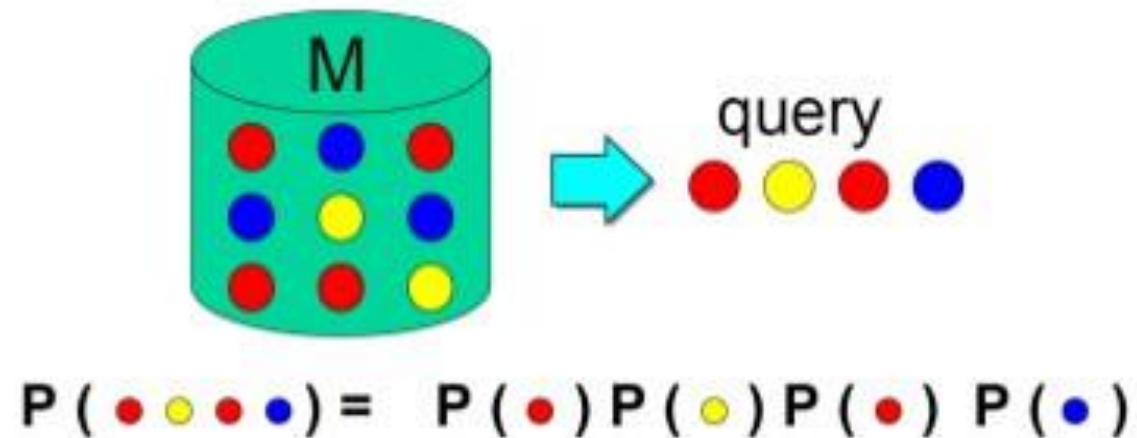
- General language modeling tasks
 - Building a unique model from a large corpus of text
 - Calculating the probability of generating a string based on the language model
 - Query likelihood task
 - Building a different language model for each document in the search space (each document is treated as the basis for a language model)
 - Calculating the probability of generating the query q based on the language model of each document d (probability that a user who likes D would pose query Q)
 - $P(Q|D_1), P(Q|D_2), \dots, P(Q|D_n)$ OR $P(Q|M_{D1}), P(Q|M_{D2}), \dots, P(Q|M_{Dn})$
- => Ranking documents based on the probabilities

Computing Query Likelihood Probability

- Simplification assumption: Unigram Model
 - Words are sampled independently
 - Order of the words is not taken into the consideration (no phrases)

$$P(Q|D) = \prod_{i=1}^n P(q_i|D)$$

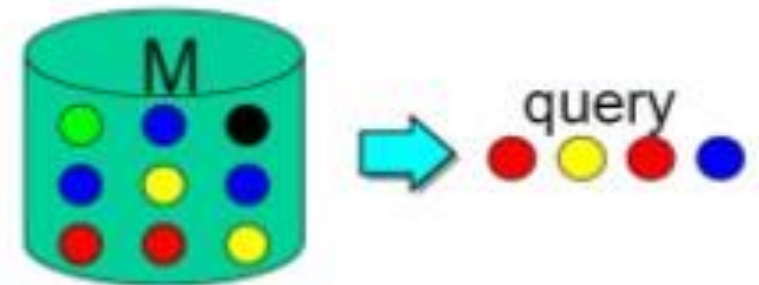
- $n = |q|$: length of q
- q_i : the token occurring at position i in q



Multinomial vs Multiple-bernoulli

- Multinomial model
 - Predominant model
 - Fundamental event: what is the identity of the i 'th query token?
 - Observation is a sequence of events, one for each query token

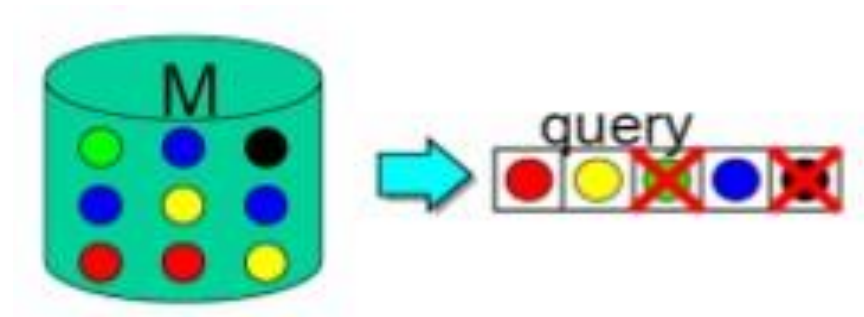
$$P(q_1 q_2 \dots q_k | M_D) = \prod_{i=1}^k P(q_i | M_D)$$



Multinomial vs Multiple-bernoulli

- Multiple-bernoulli model
 - Original model
 - Fundamental event: does the word w occur in the query?
 - Observation is a vector of binary events, one for each possible word

$$P(q_1 q_2 \dots q_k | M_D) = \prod_{w \in q_1 q_2 \dots q_k} P(w | M_D) \prod_{w \notin q_1 q_2 \dots q_k} (1 - P(w | M_D))$$



Parameter Estimation

- Main part: calculating parameters $P(w|M_D)$
- Using maximum likelihood estimates (as is used in Naïve Bayes classifiers)
- $\hat{P}(w|M_D) = \frac{TF_{w,D}}{|D|}$
 - $|D|$: length of D ;
 - $TF_{w,D}$: # occurrences of w in D

Higher Order LM

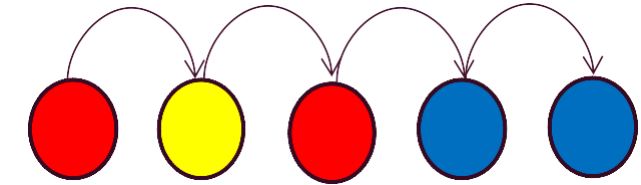
- Unigram model assumes word independence
 - Cannot capture surface form: $P(\textit{“brown dog”}) \neq P(\textit{“dog brown”})$
- Higher-order models
 - N-grams
 - Skip n-grams
 - Dependency relations

Higher Order LM

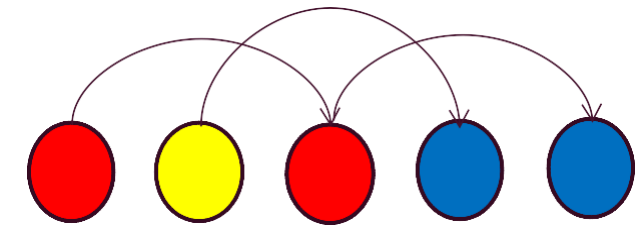
- N-grams: condition on preceding words

- Bigram $P(Q|D) = P(q_1|D) \prod_{i=2}^n P(q_i | q_{i-1}, D)$

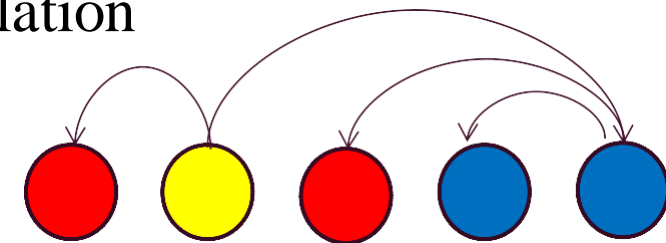
- Trigram: $P(Q|D) = P(q_1|D)P(q_2|D) \prod_{i=3}^n P(q_i | q_{i-2}q_{i-1}, D)$



- Skip n-grams: condition on previous words but not adjacent ones



- Dependency relations: condition on words with a grammatical relation



Bigrams estimation

- The maximum likelihood Estimate

- $P(Q|D) = P(q_1|D) \prod_{i=2}^n P(q_i | q_{i-1}, D)$

- $P(q_i | q_{i-1}, D) = \frac{\text{Count}_D(q_{i-1}, q_i)}{\text{Count}_D(q_{i-1})} = \frac{C(q_{i-1}, q_i)}{C(q_{i-1})}$

Bigrams estimation-Example I

- $P(q_i | q_{i-1}, D) = \frac{\text{Count}_D(q_{i-1}, q_i)}{\text{Count}_D(q_{i-1})} = \frac{C(q_{i-1}, q_i)}{C(q_{i-1})}$

D:

<s> I am Sam </s>

<s> Sam I am </s>

<s> I do not like green eggs and ham </s>

$$\begin{array}{lll} P(\text{I} | \text{<s>}) = \frac{2}{3} = .67 & P(\text{Sam} | \text{<s>}) = \frac{1}{3} = .33 & P(\text{am} | \text{I}) = \frac{2}{3} = .67 \\ P(\text{</s>} | \text{Sam}) = \frac{1}{2} = 0.5 & P(\text{Sam} | \text{am}) = \frac{1}{2} = .5 & P(\text{do} | \text{I}) = \frac{1}{3} = .33 \end{array}$$

Bigrams estimation-Example I I

Doc: Berkeley restaurant

- can you tell me about any good cantonese restaurants close by
- mid priced thai food is what i'm looking for
- tell me about chez panisse
- can you give me a listing of the kinds of food that are available
- i'm looking for a good place to eat breakfast
- when is caffe venezia open during the day
- ...

9222 sentences.

Bigrams estimation-Example 11

	i	want	to	eat	chinese	food	lunch	spend
i	5	827	0	9	0	0	0	2
want	2	0	608	1	6	6	5	1
to	2	0	4	686	2	0	6	211
eat	0	0	2	0	16	2	42	0
chinese	1	0	0	0	0	82	1	0
food	15	0	15	0	1	4	0	0
lunch	2	0	0	0	0	1	0	0
spend	1	0	1	0	0	0	0	0

$$C(q_{i-1}, q_i) = \text{Count}(q_{i-1}, q_i) \text{ in } D$$

Bigrams estimation-Example I I

- Normalize by unigrams

i	want	to	eat	chinese	food	lunch	spend
2533	927	2417	746	158	1093	341	278

- Results: $P(q_i|q_{i-1}, D) = \frac{C(q_{i-1}, q_i)}{C(q_{i-1})}$

	i	want	to	eat	chinese	food	lunch	spend
i	0.002	0.33	0	0.0036	0	0	0	0.00079
want	0.0022	0	0.66	0.0011	0.0065	0.0065	0.0054	0.0011
to	0.00083	0	0.0017	0.28	0.00083	0	0.0025	0.087
eat	0	0	0.0027	0	0.021	0.0027	0.056	0
chinese	0.0063	0	0	0	0	0.52	0.0063	0
food	0.014	0	0.014	0	0.00092	0.0037	0	0
lunch	0.0059	0	0	0	0	0.0029	0	0
spend	0.0036	0	0.0036	0	0	0	0	0

Bigrams estimation-Example 11

- $P(\langle s \rangle \text{ I want English food } \langle /s \rangle)$

$P(\text{I} | \langle s \rangle)$

× $P(\text{want} | \text{I})$

× $P(\text{English} | \text{want})$

× $P(\text{food} | \text{English})$

× $P(\langle /s \rangle | \text{food})$

=.000031

Zero Probability Problem

$$P(Q|D) = \prod_{i=1}^n P(q_i|D)$$

- A single q with $P(q|D) = 0$ will make $P(Q|D) = \prod P(q|D)$ zero
- We would give a single term “veto power”
- Example:
 - For the input query [Beethoven top hits] a document about “top songs” (but not using the word “hits”) would have $P(q|D) = 0$!!!!
- We need to smooth the estimates to avoid zeros