Topical Language Models

An Overview of Estimation Techniques

Victor Lavrenko

Department of Computer Science

University of Massachusetts, Amherst

© Victor Lavrenko, Aug. 2002

Overview

1. Introduction to Language Models
2. Estimation of Language Models
3. Smoothing techniques
4. Mixture models

© Victor Lavrenko, Aug. 2002

Part 1: Introduction

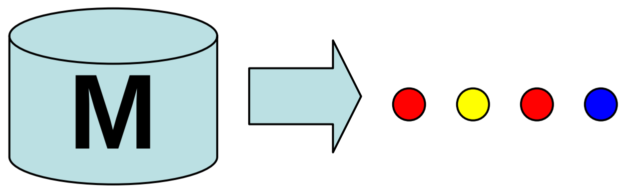
- What is a Language Model?
  - A statistical model for generating text
  - Unigram and higher-order models
  - The fundamental problem of Language Modeling
- Applications of language models
  - Information Retrieval
  - Topic Detection and Tracking
  - Question Answering / Summarization
  - Speech Recognition / Machine Translation

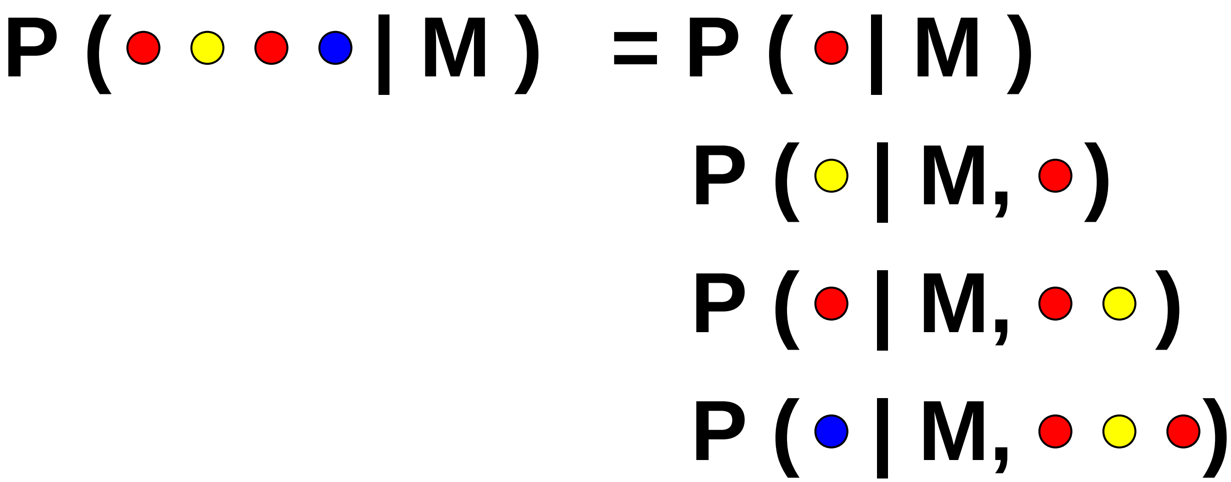
  …

© Victor Lavrenko, Aug. 2002

What is a Language Model?

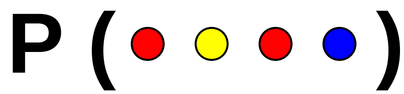
- A statistical model for generating text
  - Probability distribution over strings in a given language

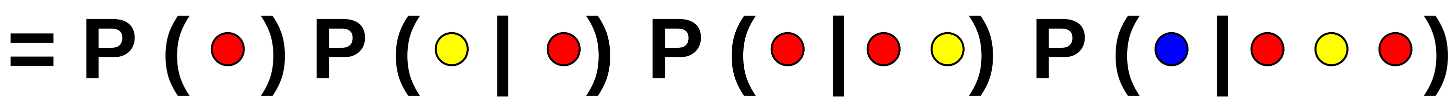




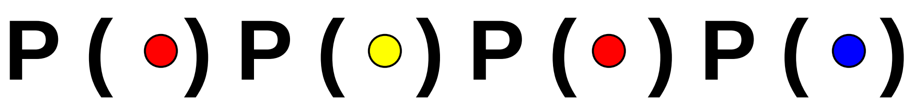
© Victor Lavrenko, Aug. 2002

Unigram and higher-order models





- Unigram Language Models



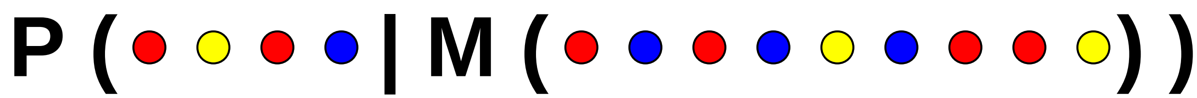
- N-gram Language Models

- Other Language Models
  - Grammar-based models, etc.

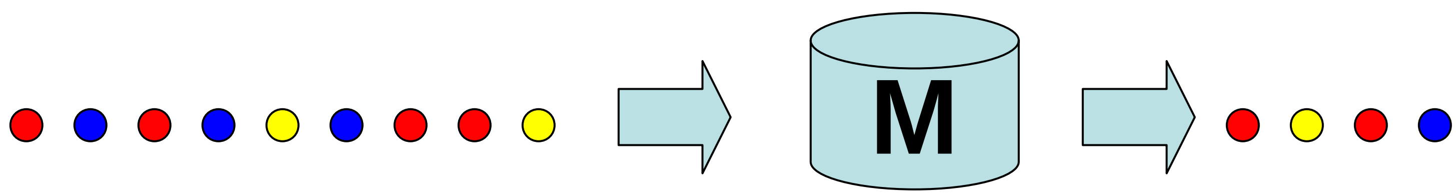
© Victor Lavrenko, Aug. 2002

The fundamental problem of LMs

- Usually we don't know the model **M**
  - But have a sample of text representative of that model



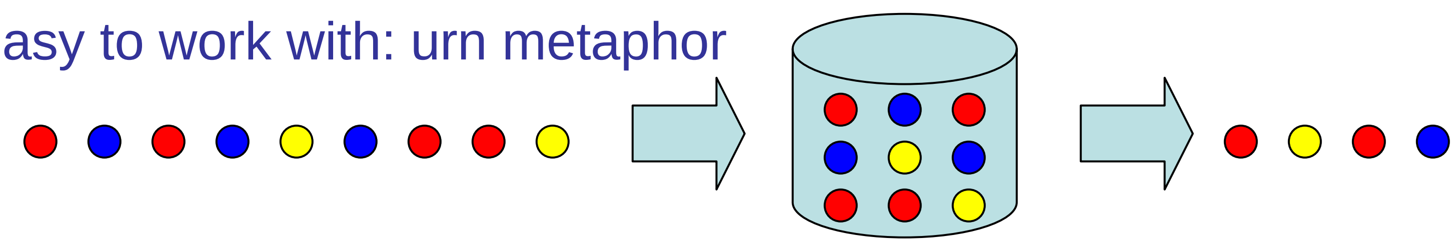
- Estimate a language model from a sample
- Then compute the observation probability



© Victor Lavrenko, Aug. 2002

Will Focus on Unigram Models

- Claim: higher-order models not necessary
  - Focus on surface form of text (well-formedness, not meaning)
  - Parameter space is too large to estimate from small samples
- Unigram models are sufficient
  - Relatively easy to estimate
  - Effective in various IR applications
  - Very easy to work with: urn metaphor



P (● ● ● ●) ~ P (●) P (●) P (●) P (●)

= 4 / 9 * 2 / 9 * 4 / 9 * 3 / 9

© Victor Lavrenko, Aug. 2002

So what's new here?

- LMs very similar to classical models of IR
  - But there are important distinctions
- Slightly different probability spaces:
  - Classical models focus on frequency space
  - Language models focus on vocabulary space
- No notions of "relevance", "user"
  - Replaced by a simple formalism
- Restricted choice of estimation methods
  - Pretty-much stuck with the "urn" metaphor
  - A lot of well-studied statistical estimation techniques

© Victor Lavrenko, Aug. 2002

Applications: Information Retrieval

- General idea
  - Estimate a language model from a document
  - Rank models by probability of “pulling out” the query
- Assumptions
  - Idea of “Relevance” replaced by “sampling”
  - Distinct language model for every document
- Multiple-Bernoulli Model
  - Ponte & Croft
- Multinomial Models
  - Berger & Lafferty, Miller et al, Hiemstra et al, …

© Victor Lavrenko, Aug. 2002

Other Applications

- Topic Detection and Tracking
  - Estimate a topic model from a few training examples
  - Compute probabilities for observing subsequent stories
- Novelty Detection
- Question Answering
  - Estimate the desired topic model (and answer-type model)
  - Extract an answer string with highest probability
- Speech Recognition / Machine Translation
  - Tri-gram models used for surface form of text
  - Unigram models useful in capturing the topical bias
    - estimation from sparse samples comes in very handy

© Victor Lavrenko, Aug. 2002

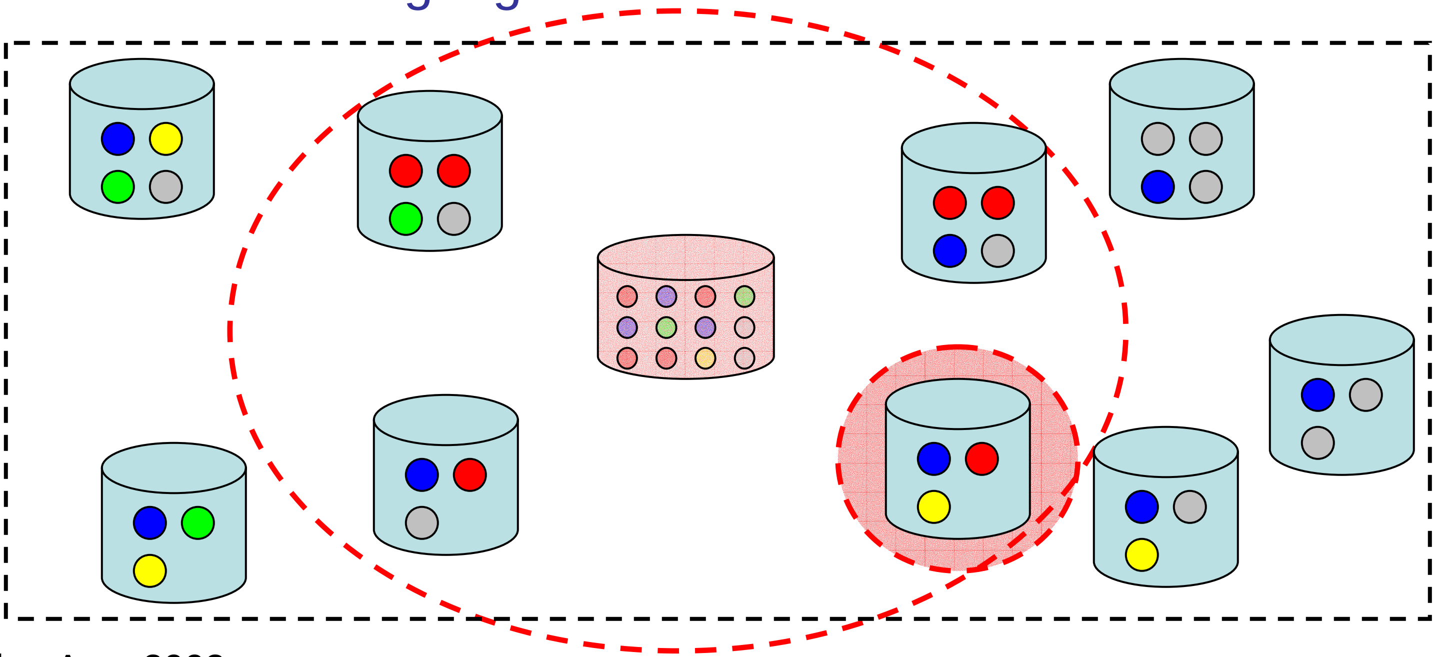
Part 2: Estimation

- Problem Statement:
  - Estimate a model from an incomplete set of examples
  - Approach: counting relative frequencies
- Properties:
  - Maximum-likelihood
  - Maximum-entropy
  - Unbiased
- Problems:
  - High-variance
  - Zero-frequency problem

© Victor Lavrenko, Aug. 2002

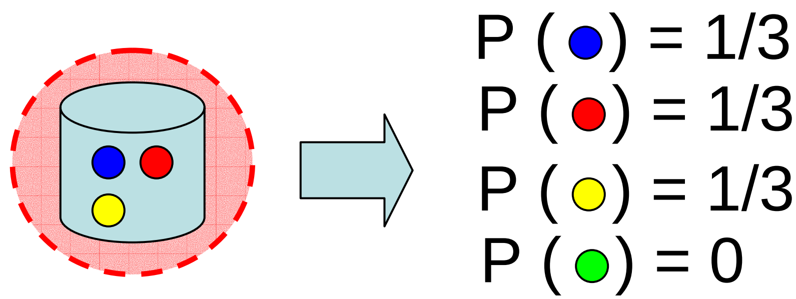
Estimation from unknown set

- Interesting models usually defined by a set
  - e.g. the set of relevant documents, or set of answers in Q/A
  - would like to estimate language model of the set
- The complete set is usually unknown
  - goal: estimate a language model from what's available



© Victor Lavrenko, Aug. 2002

Start with Maximum Likelihood

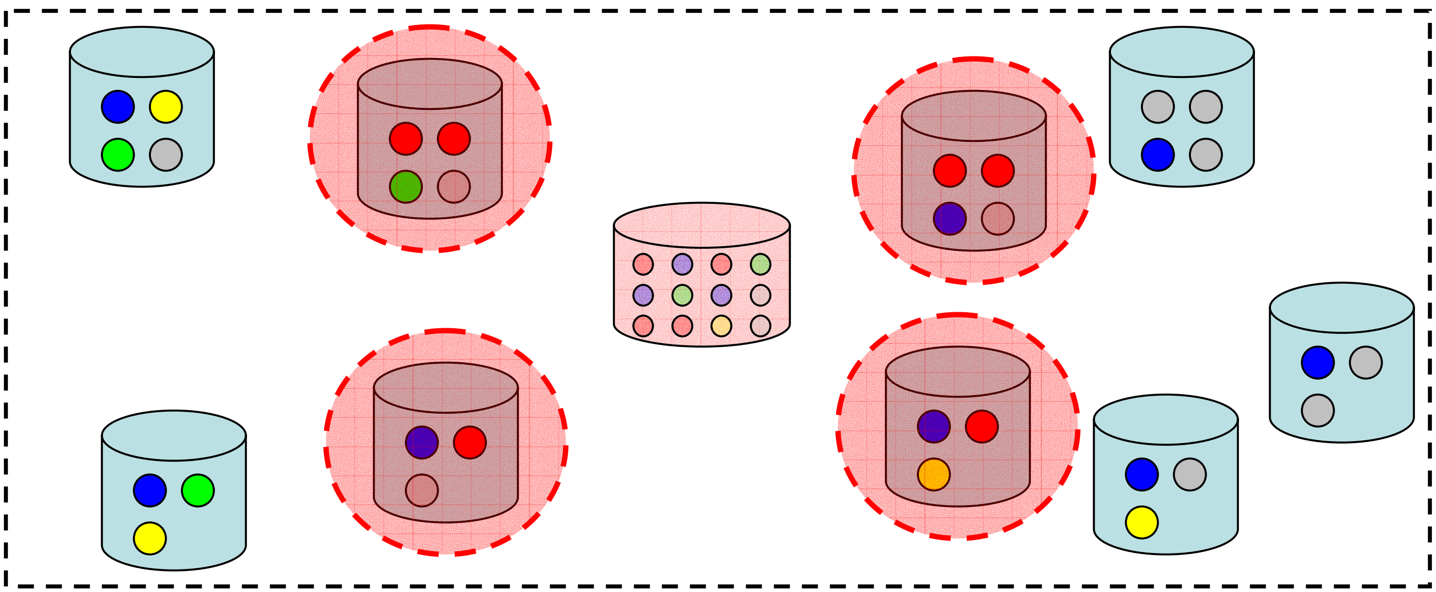


- Count relative frequencies in the example
  - hoping they would be representative of the full set
- Maximum-likelihood property:
  - resulting model gives highest probability to the example
- Maximum-entropy property:
  - resulting model makes the fewest assumptions (most random)

© Victor Lavrenko, Aug. 2002

ML Estimator is Unbiased

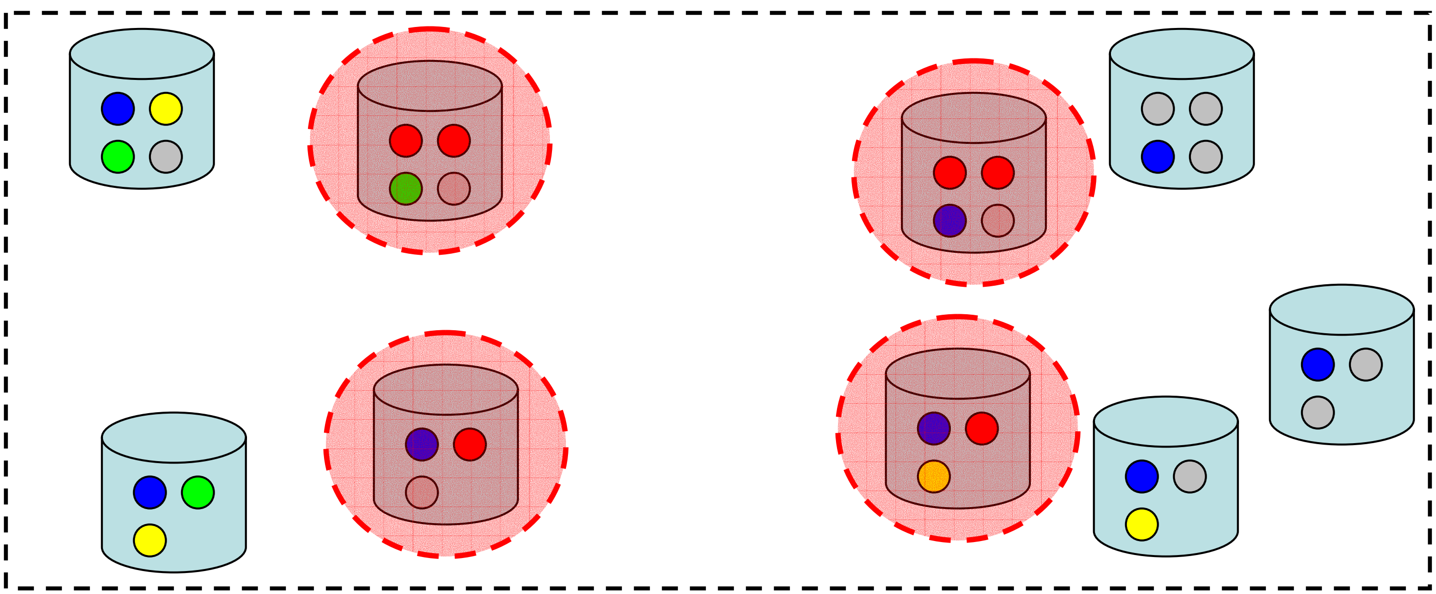
- Suppose we repeat estimation many times
- On average we get correct probabilities!
  - Expectation of the estimate has zero bias



© Victor Lavrenko, Aug. 2002

ML leads to high variance

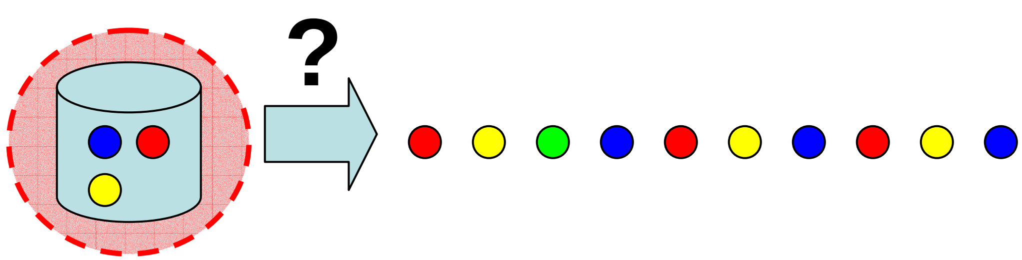
- On average, the probabilities are correct
- But there’s a serious problem:
  - Each time we can get completely different estimates!
  - Very high variance of the estimator



© Victor Lavrenko, Aug. 2002

The Zero-frequency Problem

- Suppose some event not in our example
  - Model will assign zero probability to that event
  - And to any set of events involving the unseen event
- Happens very frequently with language
- It is incorrect to infer zero probabilities
  - Especially when dealing with incomplete samples



© Victor Lavrenko, Aug. 2002

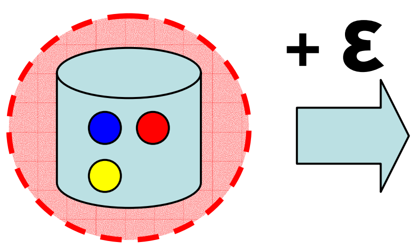
Part 3: Smoothing Techniques

- Idea:
  - Shift the probability mass towards unseen words
- Discounting Methods:
  - Laplace correction, Good-Touring, etc.
- Interpolation Methods:
  - Jelinek-Mercer, Dirichlet prior, Witten-Bell
- Automatic parameter estimation:
  - Zhai-Lafferty method
- Interpolation vs. back-off

© Victor Lavrenko, Aug. 2002

Discounting Methods

- Laplace correction:
  - Add a small constant ***ε*** to every count
- Pros:
  - Avoids zero frequencies
  - Reduces estimator variance, introduces a bias
  - ***ε*** serves as a bias-variance "tuner"
- Problem: treats all unseen events equally



P (●) = (1 + ε) / (3+5ε)

P (●) = (1 + ε) / (3+5ε)

P (●) = (1 + ε) / (3+5ε)

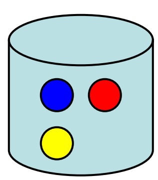
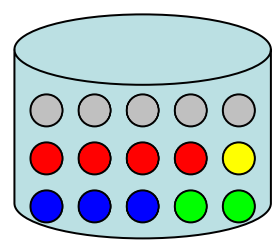
P (●) = (0 + ε) / (3+5ε)

P (●) = (0 + ε) / (3+5ε)

© Victor Lavrenko, Aug. 2002

Interpolation Methods

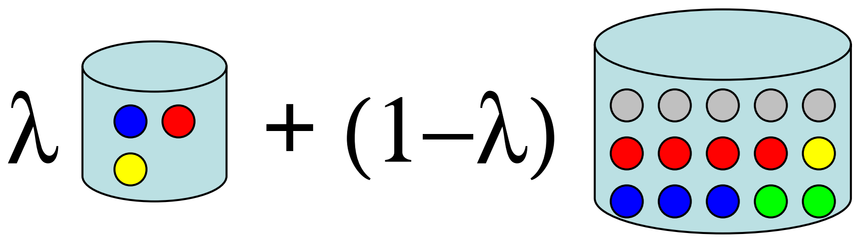
- Idea: use background (General English) probabilities for adjusting the counts
  - Reflects expected frequency of events
  - Lower bias than discounting methods, same variance
  - Smoothing parameter **λ** can serve as bias-variance tradeoff
- In IR applications, plays the role of IDF

λ  $+ (1-\lambda)$ 

© Victor Lavrenko, Aug. 2002

“Jelinek-Mercer” Smoothing

- Correctly setting **λ** is very important
- Start simple:
  - set **λ** to be a constant, independent of example
- Tune to optimize the bias-variance tradeoff

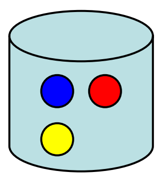


© Victor Lavrenko, Aug. 2002

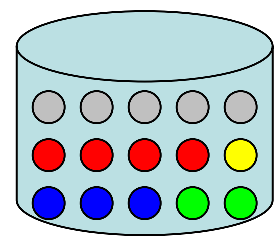
“Dirichlet” Smoothing

- Problem with Jelinek-Mercer:
  - Longer examples provide better estimates (lower variance)
  - Could get by with less smoothing (lower bias)
- Make smoothing depend on sample size
- Formal derivation
  - conjugate priors for multinomial distributions [Zhai & Lafferty ’01]

$$\underbrace{N / (N + \mu)}_{\lambda}$$



$$+ \quad \underbrace{\mu / (N + \mu)}_{(1-\lambda)}$$

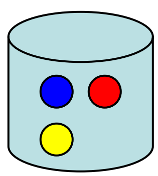


© Victor Lavrenko, Aug. 2002

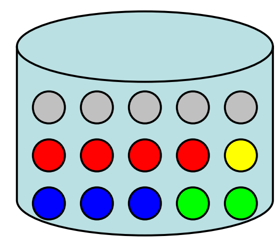
“Witten-Bell” Smoothing

- A step further:
  - Condition smoothing on “redundancy” of the example
  - Long, redundant example requires little smoothing
  - Short, sparse example requires a lot of smoothing
- Derived by considering the proportion of new events as we walk through example

$\underbrace{N / (N + V)}_{\lambda}$



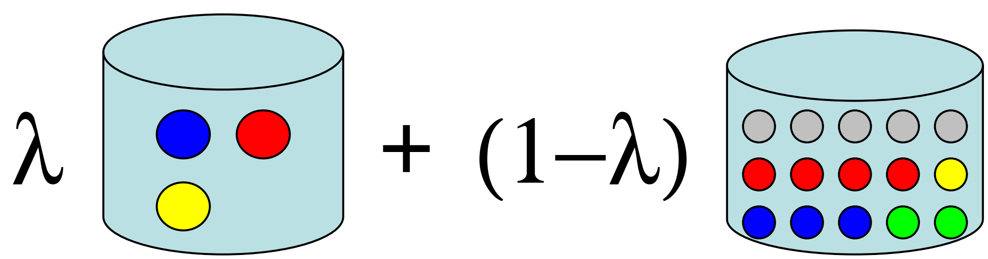
$+ \; \underbrace{V / (N + V)}_{(1-\lambda)}$



© Victor Lavrenko, Aug. 2002

"Zhai-Lafferty" Smoothing

- Leave-one-out estimation:
  - Randomly remove some word from the example
  - Compute the likelihood for the original example, based on **λ**
  - Repeat for every word in the sample
  - Adjust **λ** to maximize the likelihood
- Performs as well as well-tuned Dirichlet
  - But does not require parameter tuning



© Victor Lavrenko, Aug. 2002

Interpolation vs. back-off

- Two possible approaches to smoothing
- Interpolation:
  - Adjust probabilities for all events, both seen and unseen
- Back-off:
  - Adjust probabilities only for unseen events
  - Leave non-zero probabilities as they are
  - Rescale everything to sum to one:
    - rescales “seen” probabilities by a constant
- Interpolation tends to work better
  - And has a cleaner probabilistic interpretation

© Victor Lavrenko, Aug. 2002

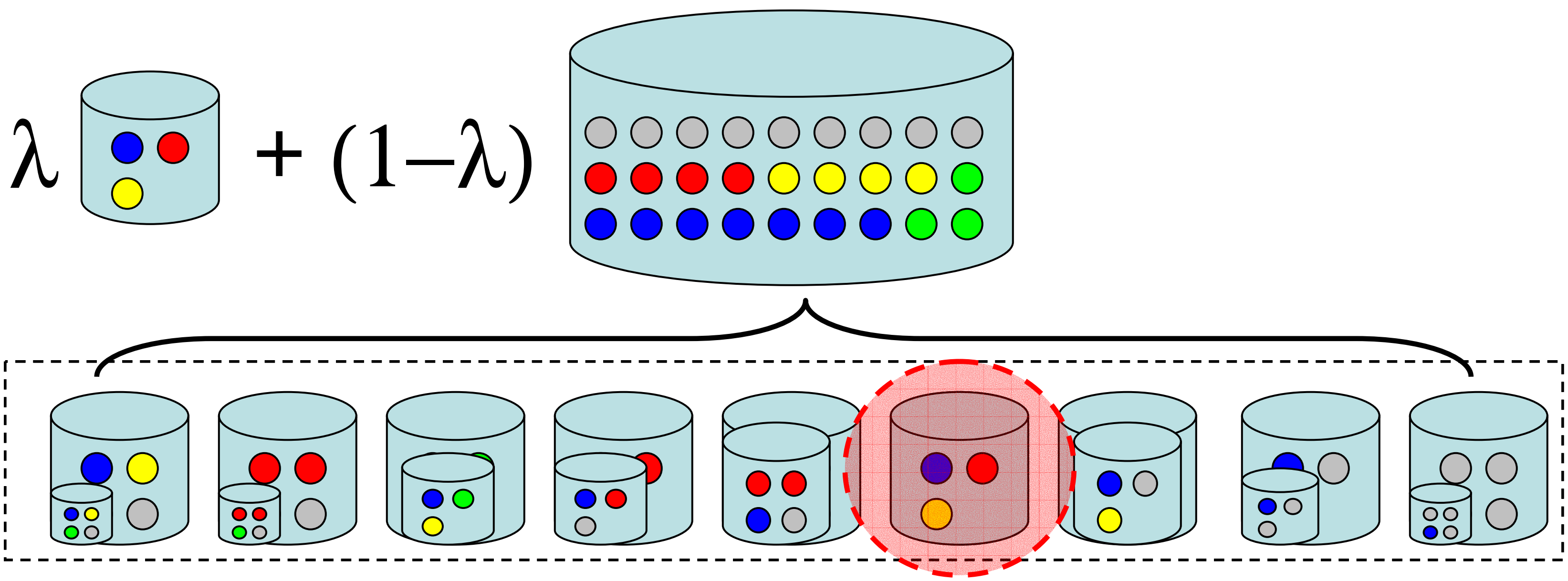
Part 4: Mixture Models

- General idea:
  - A very powerful extension of smoothing techniques
  - Allow estimation of models from extremely short samples
  - Massive Query expansion is an integral part of the model
- Probabilistic Latent Semantic Indexing
- Markov Chains on Inverted Lists
- Relevance-based Language Models
- Optimal Mixture Models

© Victor Lavrenko, Aug. 2002

Mixture Models: General Idea

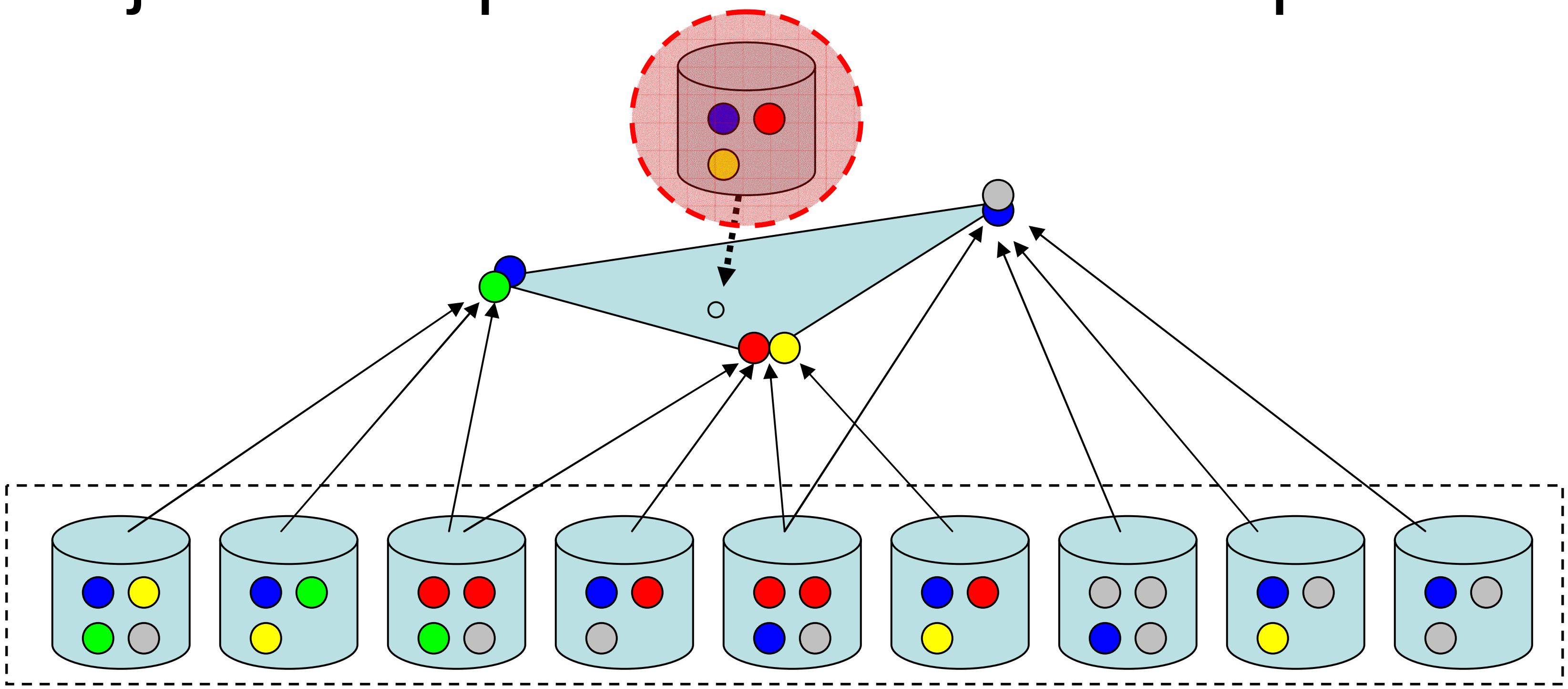
- Smoothing is a primitive mixture model
  - General English is a uniform mixture of all docs in the collection
- Consider non-uniform mixtures of docs
  - Weighted by similarity to the starting example



© Victor Lavrenko, Aug. 2002

Probabilistic LSI

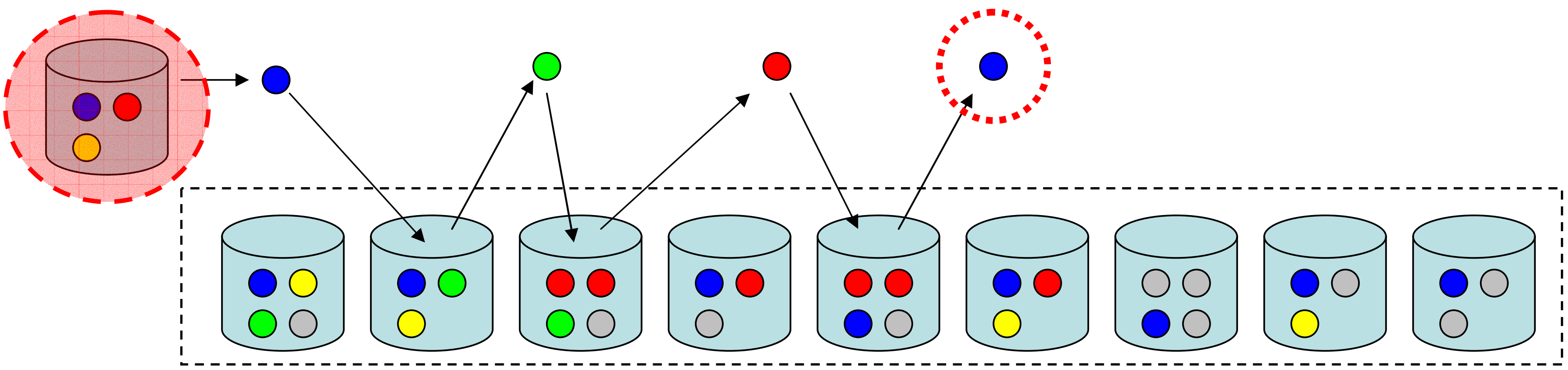
- Induce "aspects" as linear mixtures of docs
- Construct a sub-simplex with aspect basis
- Project examples onto that sub-simplex



© Victor Lavrenko, Aug. 2002

Markov Chains on Inverted Lists

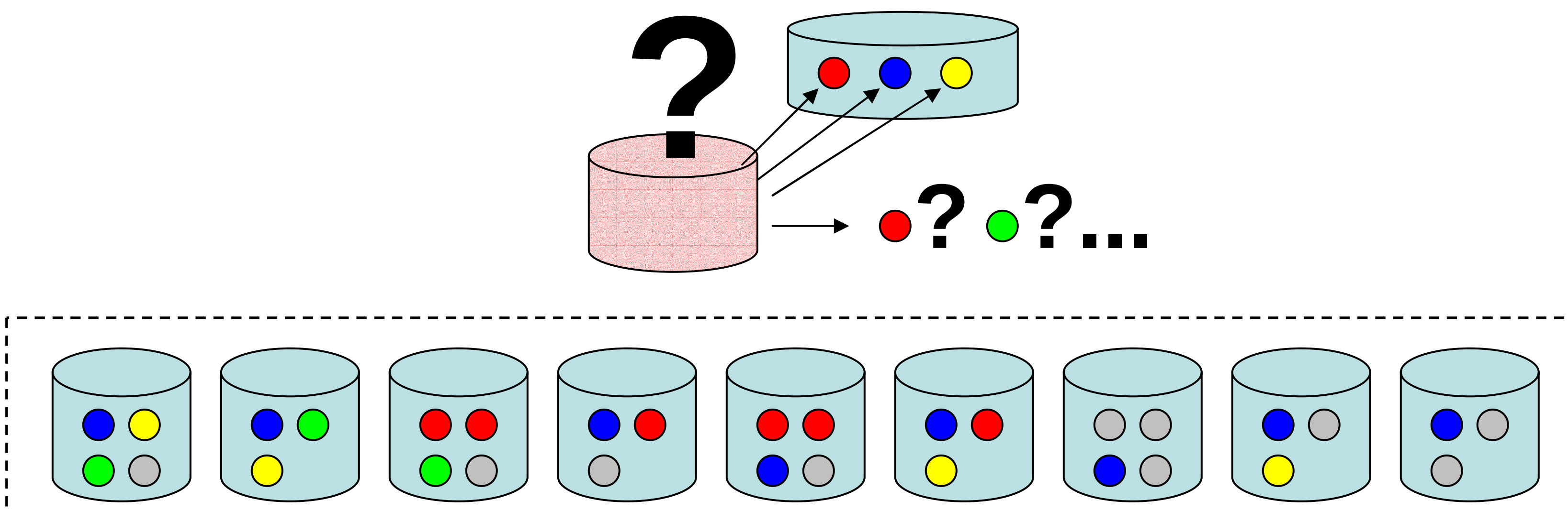
- Starting with a random word from the example
  - Pick a random doc from that word's inverted list
  - Pick a random word from that document
  - Toss ***ε*-**coin, if head – stop, else repeat
- Resulting distribution is a weighted mixture



© Victor Lavrenko, Aug. 2002

Relevance-based Models

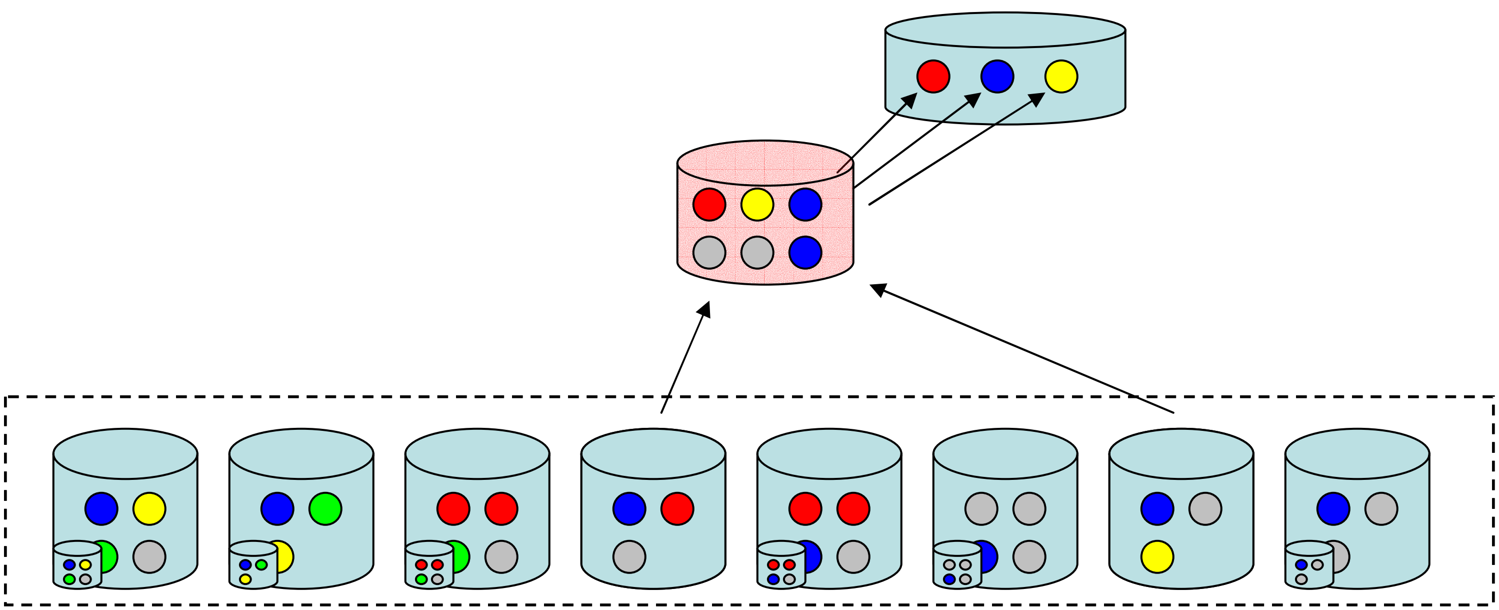
- Play a sampling game:
  - Assume there is a hidden underlying topic model
  - We sampled 3 times and observed our example:
  - What do we expect to see if we sample one more time?
  - Can compute the distribution conditioned on what we observed



© Victor Lavrenko, Aug. 2002

Optimal Mixture Models

- Extension of Relevance-based Models
  - Assume example was drawn from a **subset** of models
  - Find weighted subset that gives highest likelihood to example



© Victor Lavrenko, Aug. 2002

Summary

- Topical Language Models
  - Applications in a number of important areas
  - Principle question: estimation from incomplete examples
- Smoothing
  - A technique for reducing estimator variance
  - Discounting, interpolation, importance of smoothing parameter
- Mixture Models
  - Powerful extension of smoothing methods
  - Allows estimation from very sparse samples

© Victor Lavrenko, Aug. 2002