Language Modeling with the Maximum Likelihood Set: Complexity Issues and the Back-off Formula

Damianos Karakos
(work done jointly with Sanjeev Khudanpur)

Center for Language and Speech Processing
Johns Hopkins University

IPAM - Jan 25, 2006

What is Language Modeling?

"A probability assignment to any word sequence"

$$P(w_1, \ldots, w_m) =$$

$$P(w_1)P(w_2|w_1) \cdots P(w_m|w_{m-1}, \ldots, w_1)$$

What is Language Modeling?

"A probability assignment to any word sequence"

$$P(w_1, \ldots, w_m) =$$

$$P(w_1)P(w_2|w_1) \cdots P(w_m|w_{m-1}, \ldots, w_1)$$

$$\approx P(w_1)P(w_2|w_1) \cdots P(w_m|w_{m-1}, \ldots, w_{m-N+1})$$

(N-gram approximation)

See the book: **Fred Jelinek**, "Statistical Methods for Speech Recognition", MIT Press

Where is it used?

- Document Categorization
- $P(\text{document}|\text{category})$
- Information Retrieval
- $P(\text{query}|\text{document})$ ← more from D. Hiemstra
- Speech Recognition

$$\boldsymbol{W}^* = \arg\max_{\boldsymbol{W}} P(\boldsymbol{W}|A) = \arg\max_{\boldsymbol{W}} P(A|\boldsymbol{W})\, P(\boldsymbol{W})$$

Is it a hard problem?

Unigram estimation: $P(w),\ \forall\, w \in \mathcal{V}$

Typically ≈ 50,000

Bigram estimation: $P(w_1|w_2),\ \forall\, (w_1, w_2) \in \mathcal{V}^2$

Trigram estimation: $P(w_1|w_2, w_3),\ \forall\, (w_1, w_2, w_3) \in \mathcal{V}^3$

It is a density estimation problem

from small samples!

Density Estimation

- Let X be a random variable taking values in some set $\{1, \ldots, k\}$
- Let P be the true distribution of X
- We wish to estimate P from a small number of samples $\{x_1, \ldots, x_n\}$

Density Estimation

- Maximum-Likelihood Estimate:

$$\hat{P}(a) = \frac{1}{n} \sum_{i=1}^{n} \mathbf{1}(x_i = a), \quad a \in \{1, \ldots, k\}$$

Data sparseness:
most values of $\hat{P}(a)$ are zero!

bad!

(particularly when $n << k$)

Solution to Data Sparseness: Smoothing

Idea: give probability mass to infrequent words by discounting the more frequent ones.

For example:

Add-one smoothing: $p_{\text{Bayes}}(a) = \dfrac{1 + c_a}{n + |V|}$

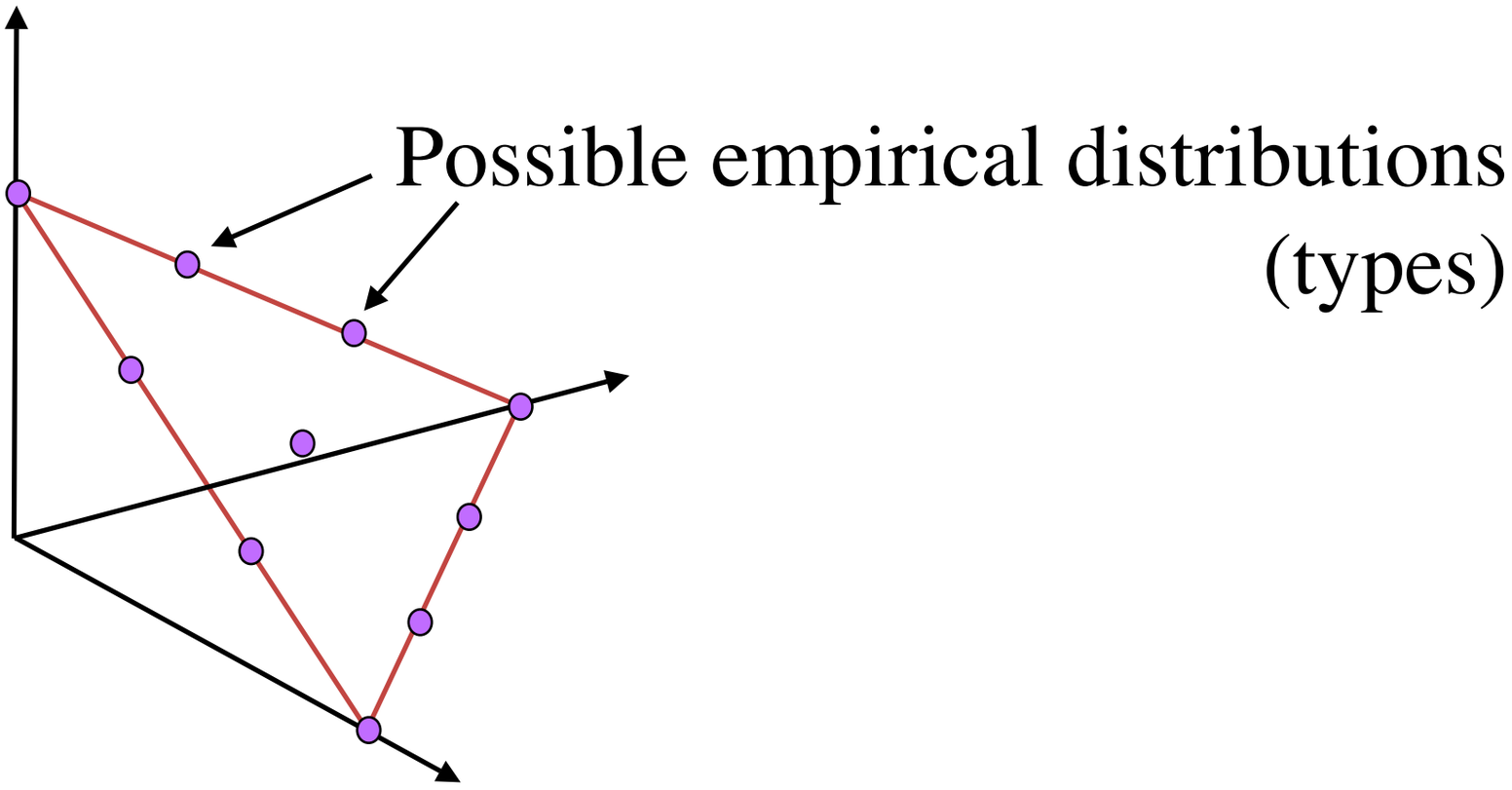
Good-Turing: $p_{GT}(a) = \dfrac{1}{n} \dfrac{(c_a + 1) n_{c_a+1}}{n_{c_a}}$

A New Approach:
The Maximum Likelihood Set

Introduced by Jedynak and Khudanpur,
Neural Computation, 2005.

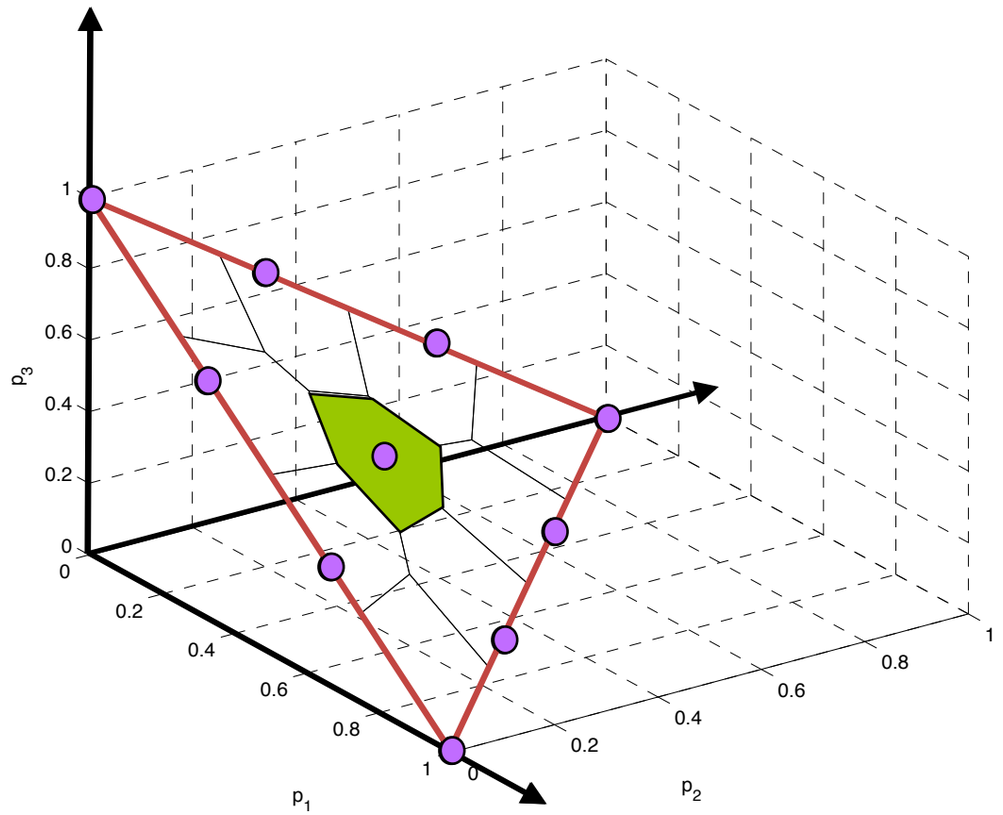
Idea: given some observed counts, MLS contains all distributions under which the observed counts are *more* probable than any other set of counts for the same sample size.

Probability Simplex



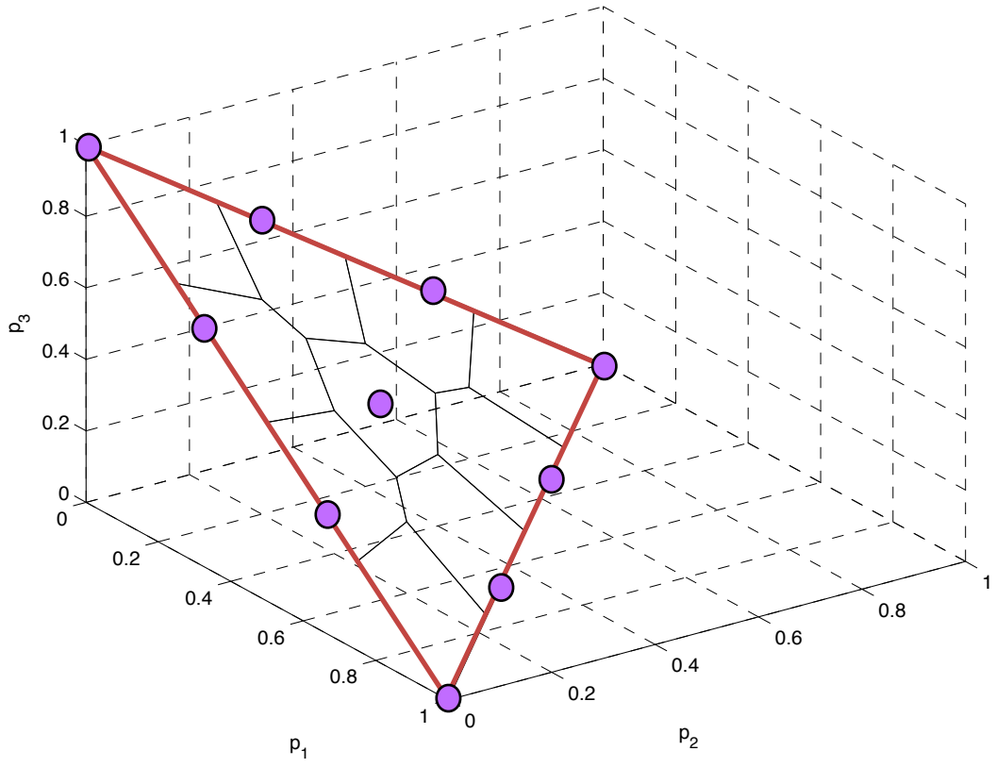
$$k = 3,\ n = 3$$

The Maximum Likelihood Set

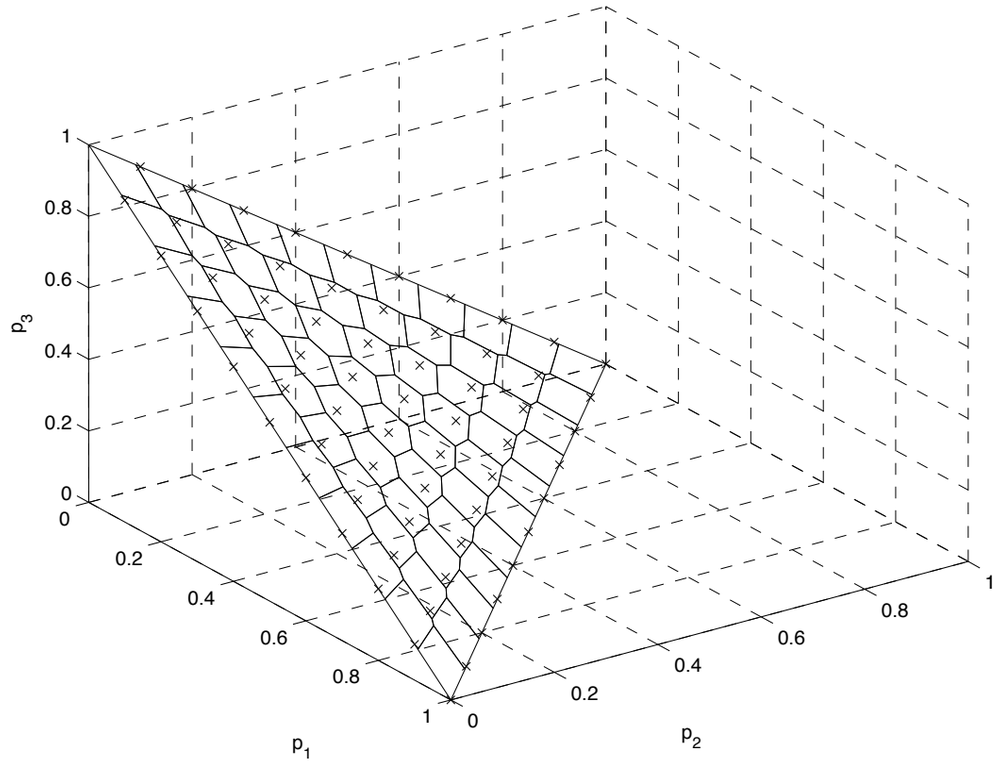


$$k = 3,\ n = 3$$

The Maximum Likelihood Set



$k = 3,\ n = 3$



$k = 3,\ n = 10$

The Maximum Likelihood Set

Formally:

For counts $c_1, \ldots, c_k$, $\sum_{i=1}^{k} c_i = n \qquad (k = |V|)$

we have

$$p(c_1, \ldots, c_k) = \frac{n!}{c_1! \ldots c_k!} \prod_{i=1}^{k} [p(i)]^{c_i}$$

$\left(\begin{array}{l}\text{Probability of observed counts under some}\\ \text{distribution } p\end{array}\right)$

The Maximum Likelihood Set

$$\mathcal{M}(c_1, \ldots, c_k) =$$

$$\left\{ p \in \mathcal{P}^k : p(c_1, \ldots, c_k) \geq p(c'_1, \ldots, c'_k),\ \forall c'_1, \ldots, c'_k : \sum_{i=1}^{k} c'_i = n \right\}$$

The set of counts we observed should be at least as likely as the ones we didn't.

The Maximum Likelihood Set

$$\mathcal{M}(c_1, \ldots, c_k) =$$

$$\left\{ p \in \mathcal{P}^k : p(i) \cdot c_j \leq p(j) \cdot (c_i + 1), \ \ \forall\, i, j \in \{1, \ldots, k\} \right\}$$

The set of counts we observed should be at least as likely as the ones we didn't.

The Maximum Likelihood Set

- Faithfulness to evidence:
  - if $c_i < c_j$ then $p(i) \leq p(j)$
- If $c_i > 0$ then $p(i) > 0$ for every p in the MLS.
- Almost every pmf in the MLS is *"smooth"*:
- $\forall\, p \in \mathcal{M},\ p(i) > 0, \forall\, i$ (except possibly for a set of Lebesgue measure zero)
- Every pmf in the MLS is a strongly consistent estimate of the true distribution.

Choosing an estimate from the MLS

- The MLS is a closed, convex polyhedron.
- If there is a reference distribution q (e.g., one that we would use when $n = 0$), then

$$p^* = \arg \min_{p \in \text{MLS}} D(p||q)$$

if $c_i = c_j$ and $q_i > q_j$ then $p_i^* > p_j^*$

Faithfulness to prior knowledge

Bigram and Trigram Estimation

$$P(w_1, \ldots, w_m) = \prod_{i=1}^{m} P(w_i | w_{i-1})$$

$$P(w_1, \ldots, w_m) = \prod_{i=1}^{m} P(w_i | w_{i-1}, w_{i-2})$$

- We have a different MLS for each history.
- Each MLS has at most $|V|^2$ constraints.
- But it is not necessary to solve thousands of convex optimizations, each of millions of constraints!

Bigram and Trigram Estimation

Two crucial sources of computational savings:

1) if $c_i = c_j$ and $q_i = q_j$, then $p_i^* = p_j^*$

$\Rightarrow$ Grouping of words together

Vocabulary size: 50,000

$\rightarrow$ effective alphabet size: ~ 600

Bigram and Trigram Estimation

Two crucial sources of computational savings:

2) Zero-count words impose constraints only to the *seen* words.

Hence, if only $z << |V|$ words are seen following a history, the number of constraints is $O(z^2) + O(z|V|) + O((|V| - z)^2)$

Bigram and Trigram Estimation

Two crucial sources of computational savings:

2) Zero-count words impose constraints only to the *seen* words.

Hence, if only $z << |V|$ words are seen following a history, the number of constraints is $O(z^2) + O(z|V|)$

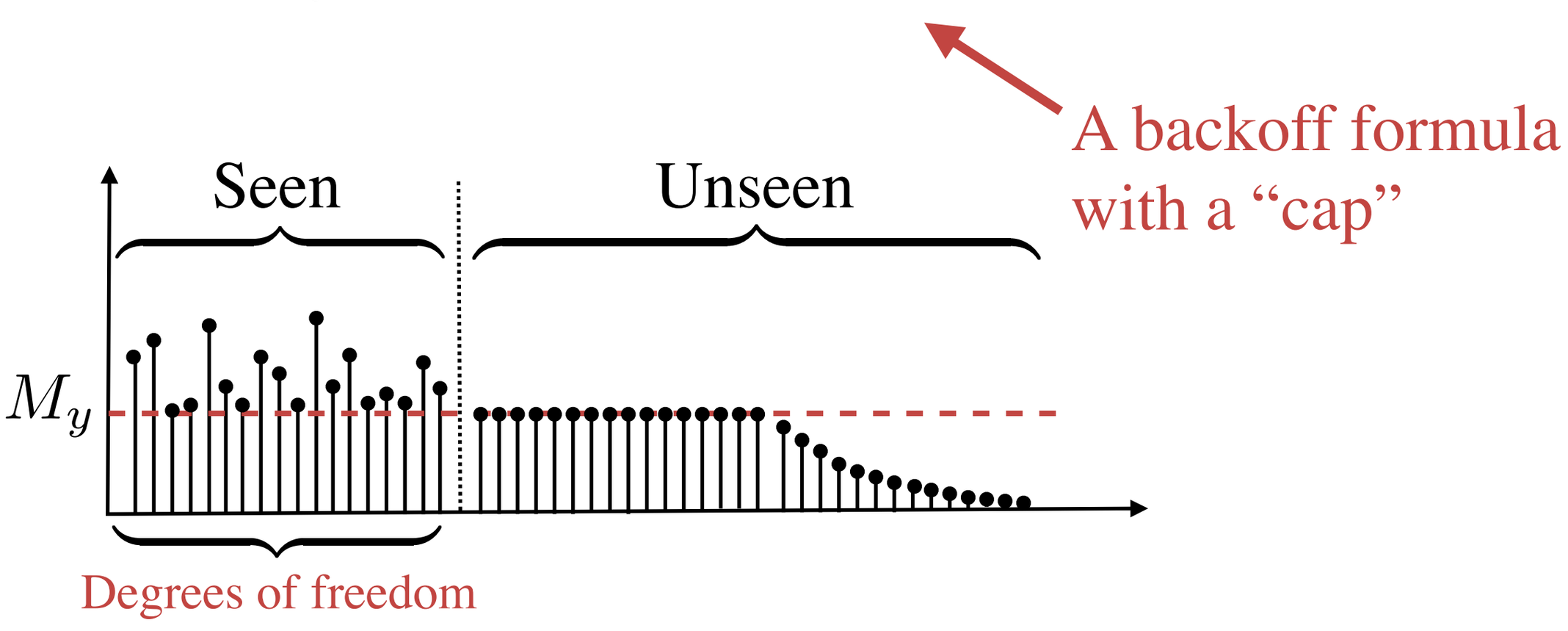
$z \approx 4$ on average

(for bigram estimation)

Model Complexity

The estimate of the conditional pmf has the form:

$$p(x|y) = \begin{cases} p^*(x|y) & \text{if } c(xy) > 0 \\ \min\{\lambda_y q(x), M_y\} & \text{if } c(xy) = 0 \end{cases}$$



Experimental Results

- Experiments on English text from the UPenn Treebank corpus (part of WSJ).
  - Training on Sec. 00-22 (≈ 1M words).
  - Testing on Sec. 23-24 (100K words).
- Vocabulary: 52,000 words (including 15,000 “unknown” words).
- Reference distributions used:
  2-gram and 3-gram Good-Turing, Witten-Bell, Kneser-Ney

Experimental Results

Avg. log-likelihood: $LL = \frac{1}{n_{\text{test}}} \sum_{i=1}^{n_{\text{test}}} \log(P_{\text{LM}}(w_i|w_{i-1}))$

Perplexity: 2^{-LL}

We report:

Total perplexity	
Perplexity on seen N-grams	Perplexity on unseen N-grams

Experimental Results

Bigram estimation

Model		Good-Turing		Witten-Bell		Modified Kneser-Ney	
Total Perplexity		415		357		329	
Seen bigrams	Unseen bigrams	78	45376	80	24106	77	19651

MLS

Total Perplexity		**411**		**348**		336	
Seen bigrams	Unseen bigrams	79	**43169**	**77**	**24012**	87	**15171**

Experimental Results

Trigram estimation

Model		Good-Turing		Witten-Bell		Modified Kneser-Ney	
Total Perplexity		354		298		270	
Seen trigrams	Unseen trigrams	18	4163	19	2887	17	2649

MLS

Total Perplexity		**352**		303		280	
Seen trigrams	Unseen trigrams	19	**3784**	20	**2787**	20	**2489**

Can we expand the MLS?

The requirement

The set of counts we observed should be at least as likely as the ones we didn't.

may be too restrictive.

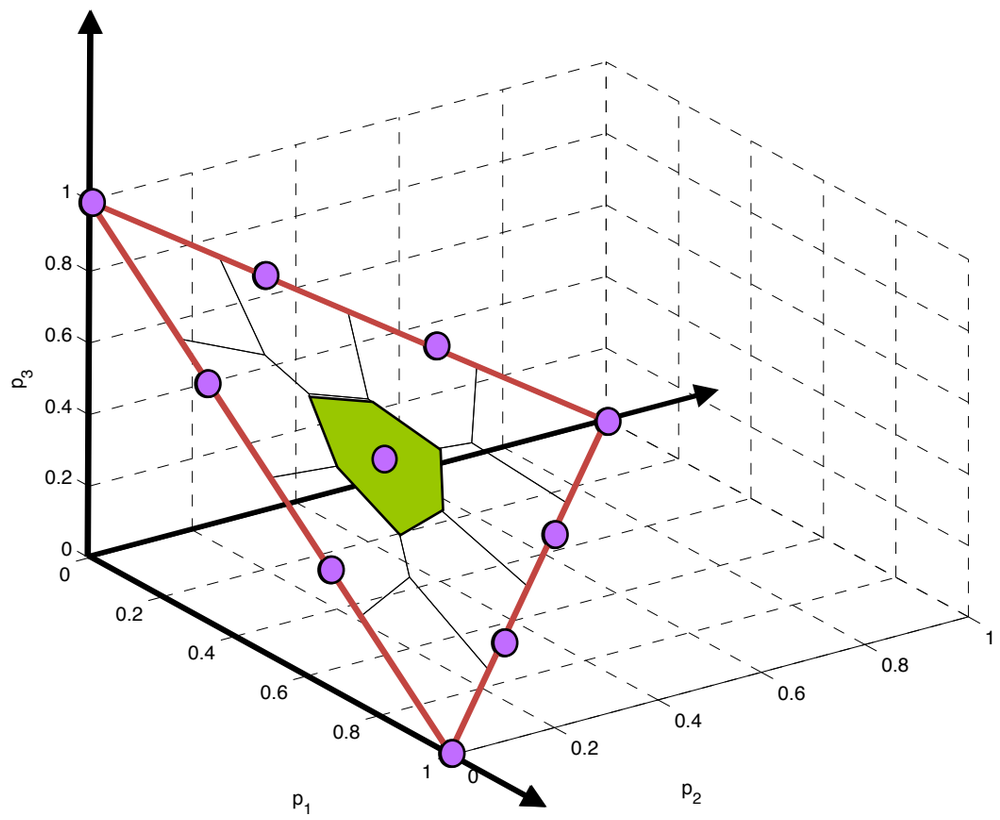
Can we expand the MLS to include more distributions?

The High Likelihood Set

$$\mathcal{H}(c_1, \ldots, c_k) =$$

$$\left\{ p \in \mathcal{P}^k \ : \ p(c_1, \ldots, c_k) \geq \alpha \, p(c'_1, \ldots, c'_k), \ \forall \, c'_1, \ldots, c'_k \ : \ \sum_{i=1}^{k} c'_i = n \right\}$$

$$0 \leq \alpha \leq 1$$

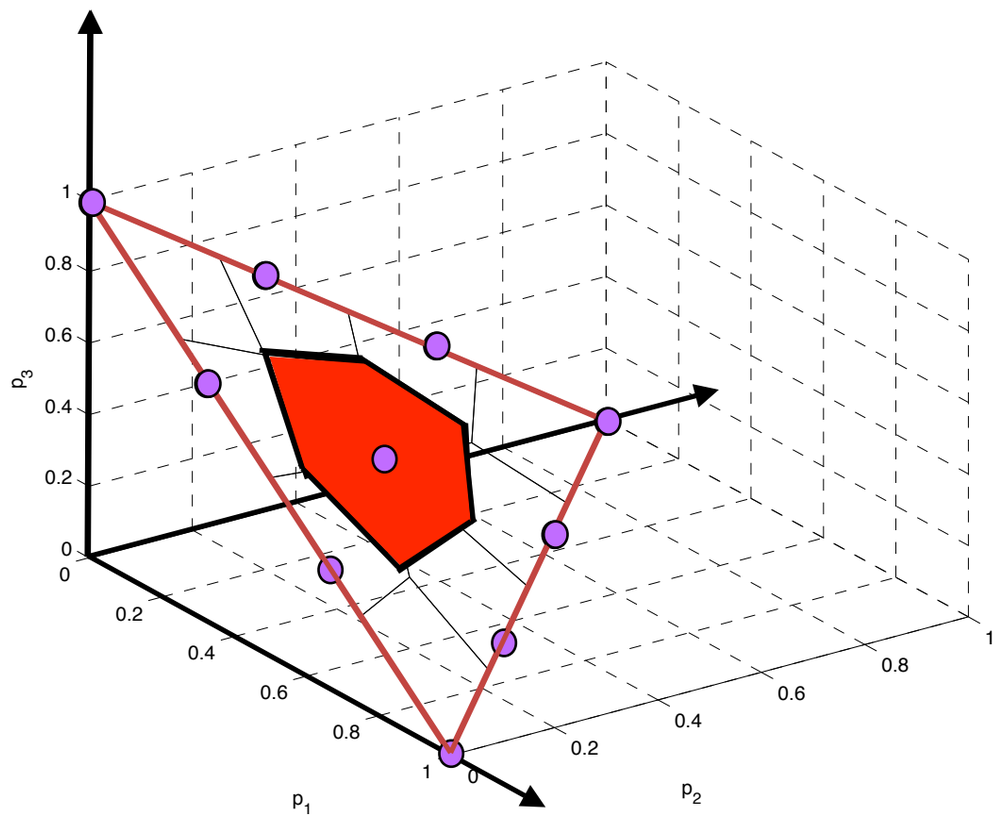


The High Likelihood Set

$$\mathcal{H}(c_1, \ldots, c_k) =$$

$$\left\{ p \in \mathcal{P}^k \ : \ p(c_1, \ldots, c_k) \geq \alpha \, p(c'_1, \ldots, c'_k), \ \forall \, c'_1, \ldots, c'_k \ : \ \sum_{i=1}^{k} c'_i = n \right\}$$

$$0 \leq \alpha \leq 1$$



The High Likelihood Set

How to pick α?

Idea:

Diameter of MLS: $O(1/n)$

Rate of shrinkage of the high-probability set around the true distribution: $O(1/\sqrt{n})$

Hence: fatten the MLS by $\sqrt{n}$

Experimental Results

Trigram estimation

Model		Good-Turing		Witten-Bell		Modified Kneser-Ney	
Total Perplexity		354		298		270	
Seen trigrams	Unseen trigrams	18	4163	19	2887	17	2649

HLS

Total Perplexity		**341**		**292**		**269**	
Seen trigrams	Unseen trigrams	**16**	**4058**	**18**	**2877**	17	**2639**

Experimental Results

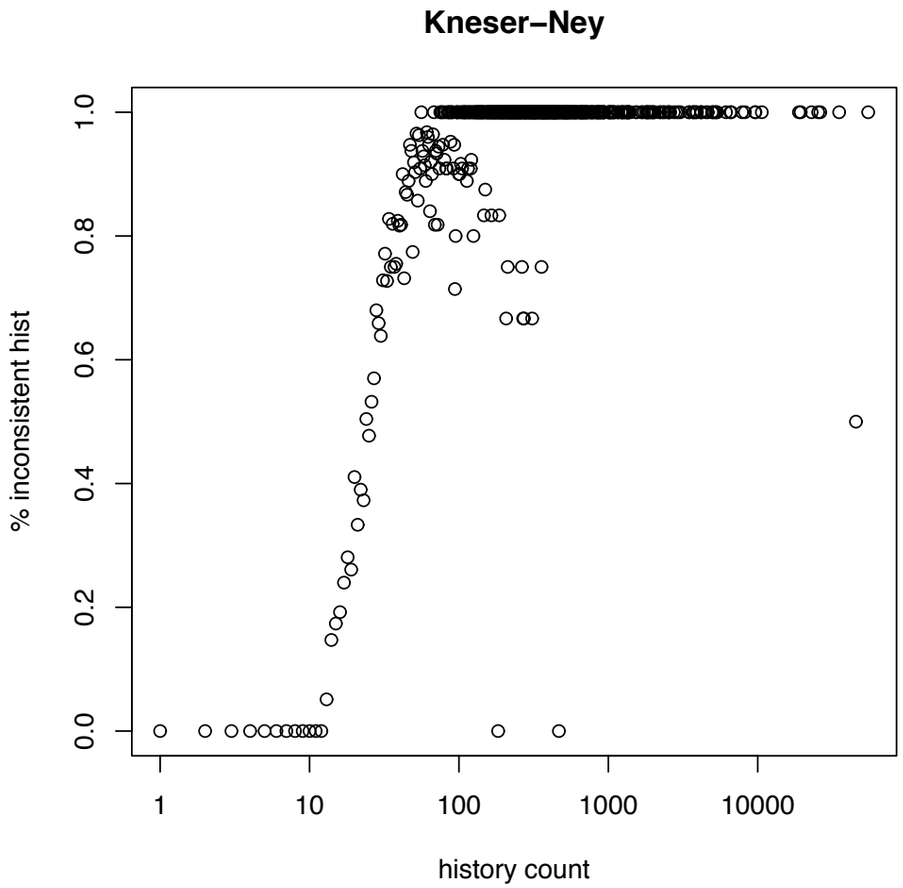
Word-Error-Rates

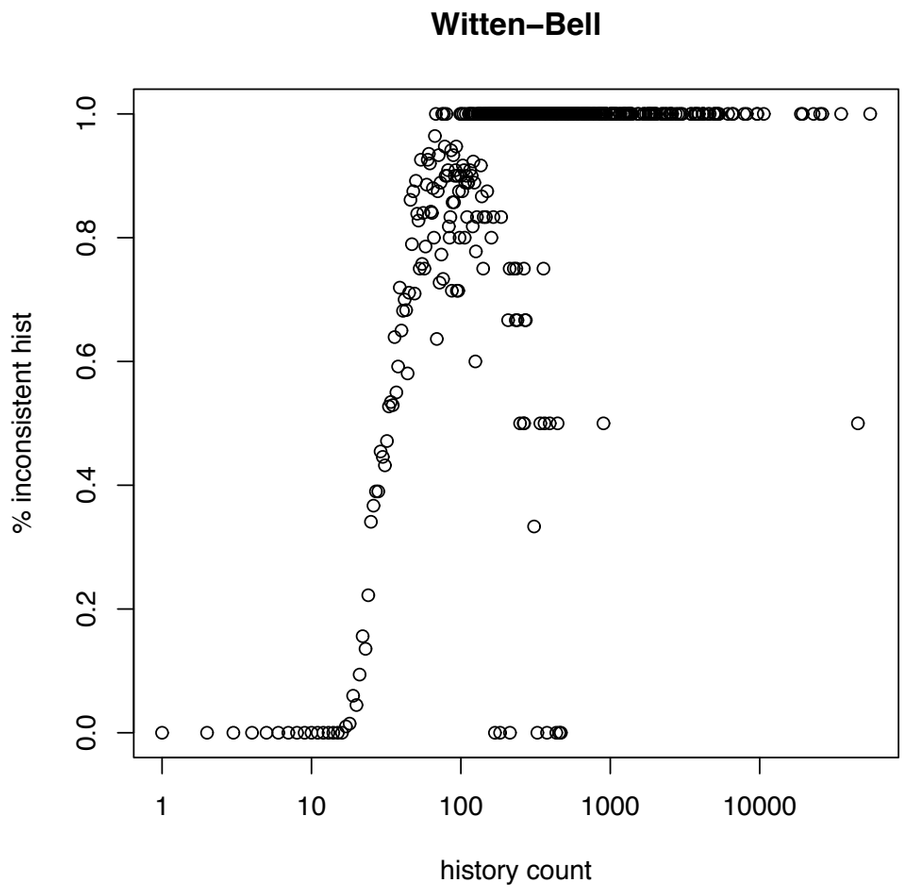
Model	Good-Turing	Witten-Bell	Modified Kneser-Ney
WER	15.7%	16.0%	15.8%

HLS

WER	**15.7%**	**15.9%**	**15.7%**

Unfaithfulness to evidence of other models





Conclusions

- The MLS contains "good" distributions, both in terms of perplexity and in terms of faithfulness to evidence.
- When the criterion is minimization of KL divergence, the estimated pmf improves the reference distribution, especially in the prediction of *unknown* words.
- "Capping" reduces the conditional probability of unseen words which are otherwise frequent.

Future Directions

- Scale to more data (100s of millions of words).
- Measure performance in other tasks (e.g., machine translation).
- Find ways of incorporating other linguistic knowledge through the reference distribution.