# Exact Maximum Likelihood Estimation for Word Mixtures


**Yi Zhang *** YIZ@CS.CMU.EDU
**Wei Xu **** XW@CCRL.SJ.NEC.COM
**Jamie Callan *** CALLAN@CS.CMU.EDU
* Language Technology Institute, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA 15213

** Multimedia Software Department, C&C Research Laboratories, NEC USA, Inc., San Jose, California, USA



## Abstract

The mixture model for generating document is a generative language model used in information retrieval. While using this model, there are situations that we need to find the maximum likelihood estimation of the density of one multinomial, given fixed mixture weight and the densities of the other multinomial. In this paper, we provide an exact solution and a quick algorithm to solve this problem. The new algorithm is guaranteed to find the exact optimal result, at a fast speed. More over, some interesting properties of the solution are discussed.


## 1. Introduction

Probabilistic language models are widely used in speech recognition and have shown promise in information retrieval (e.g., (Zhai & Lafferty, 2001)(Kraaij et al., 1999)(Zhang et al., 2002)). While using language model for IR, there are situations that we need to find the maximum likelihood estimation of the density of one multinomial, given fixed mixture weight and the densities of the other multinomial.

One example is using language modeling in ad-hoc information retrieval with relevance feedback. (Zhai & Lafferty, 2001) assume each relevant document is generated by a mixture of query model and corpus language model :

$$\theta = \alpha\theta_C + \beta\theta_{query} \tag{1}$$

where $\alpha, \beta$ are given and sum to 1. $\alpha$ indicates the amount of background "noise" when generating a document from the query model. $\theta_C$ is the multinomial distribution for the corpus language model. We can treat the maximum likelihood estimator (MLE) of $\theta_C$ as already known, since it can be calculated directly as:

$$\hat{\theta_C}_i = \hat{P}(w_i|p) = \frac{df_i}{\sum_j df_j}$$

where $df_i$ is the number of times word $i$ occurs in the whole corpus .

In this case, calculating the query model $\theta_{query}$ is the focus of the task. Intuitively, by calculating query model, noise words can be deleted, since the query model will concentrate on words occur frequently in relevant documents, but less frequent in the whole corpus.

Another example is using language modeling for redundancy/novelty detection in adaptive information filtering. (Zhang et al., 2002) assume each relevant document is generated by the mixture of three language models: A General English language model $\theta_E$ (similar to $\theta_C$), a user-specific Topic Model $\theta_T$ (similar to $\theta_{query}$), and a document-specific Information Model $\theta_{d\_core}$. Each word $w_i$ in the relevant document is generated by the mixture of the three language models with probability $\lambda_E$, $\lambda_T$ and $\lambda_{d\_core}$ respectively:

$$\begin{aligned} &P(w_i|\theta_E, \theta_T, \theta_{d\_core}, \lambda_E, \lambda_T, \lambda_{d\_core}) = \\ &\quad \lambda_E P(w_i|\theta_E) + \lambda_T P(w_i|\theta_T) + \lambda_{d\_core} P(w_i|\theta_{d\_core}) \end{aligned} \tag{2}$$

where $\lambda_E + \lambda_T + \lambda_{d\_core} = 1$.
In this case, $\lambda_E, \lambda_T, \lambda_{d\_core}$ are fixed, $\theta_E$ is calculated similar to $\theta_C$ as in formula 1, $\theta_T$ is calculated similar to $\theta_{query}$ as described before. Finding the document-specific Information Model is the focus of the novelty/redudancy detection task. Intuitively, by calculating $\theta_{d\_core}$, words does not provide new information can be deleted, since this the document-specific Information Model explicitly models "what's news" in

a document. It focuses on words occur frequently in the document, but less frequent in the whole corpus and relevant documents.

The problem needs to be solved in the above two tasks, finding query model $\theta_{query}$ or finding document information model $\theta_{d\_core}$, can be generalized as finding the maximum likelihood estimation of multinomial distribution $q$ in the following model:

$$r = \alpha p + \beta q \tag{3}$$

where $\alpha$ and $\beta$ are given, p is a known multinomial distributions, and data observed is generated according to $r$.

The traditional way to find $q$ is using the EM algorithm. For example, $\theta_{query}$ in ad-hoc retrieval task with relevance feedback can be updated as follows:

$$f_i^{(n)} = \frac{\beta q_i^{(n)}}{\alpha p_i + \beta q_i^{(n)}} \tag{4}$$

$$q_i^{(n+1)} = \theta_{query_i}^{(n+1)} = \frac{\sum_d tf_{i,d} * f_i^{(n)}}{\sum_j \sum_d tf_{i,d} * f_j^{(n)}} \tag{5}$$

where $d$ is a document relevant to the query under consideration, and $tf_{i,d}$ is the number of times word $i$ occurs in the document $d$.

Although EM based solution is used often, it has several weaknesses.

- EM can be computationally expensive, because it is a greedy search algorithm. This expense discourages the use of the language modeling approach for the information retrieval task in environments where computational efficiency is important, for example, in a large scale Web search engine. In such an environment one must choose between speed and accuracy.
- EM can only provide an approximation to the optimal result. The greater the desired accuracy, the greater the computational cost because of the iterative nature of the algorithm.

In this paper, we present an exact solution that has $k \log k$ complexity, where k is the possible values for the multinomial trials process that generates the document. We also discuss some interesting properties of the solution. In most cases the exact solution is faster than the EM algorithm; it is always more accurate. We also verify the theoretical results with experiments that find query language models, the key component of some language modeling approaches to ad-hoc information retrieval with relevance feedback (Zhai & Lafferty, 2001). As expected, the EM algorithm converges to the result calculated directly by our algorithm, and the EM algorithm is often slower than our algorithm.

## 2. Algorithm

Assume we observe a sequence of data $D$ that is assumed to be generated from a linear combination of two multinomial models $p$ and $q$. We formalize this as:

$$r = \alpha p + \beta q \tag{6}$$

where $\alpha$ and $\beta$ are interpolation weights that sum to 1. $p, q$ and $r$ are multinomial distributions. $p = (p_1, p_2, ..., p_k)$, $q = (q_1, q_2, ..., q_k)$ and $r = (r_1, r_2, ..., r_k)$. The log-likelihood of data is:

$$LL = \sum_{i=1}^{k} f_i \log(r_i) = \sum_{i=1}^{k} f_i \log(\alpha p_i + \beta q_i) \tag{7}$$

Where $f_i$ is observed frequency of item $i$(word $i$) in the data. The problem is to find $q_i$'s that maximize the likelihood of observed data $D$, for given $f_i$, $p_i$, $\alpha$ and $\beta$, subject to the constraints $\sum_i q_i = 1$ and $q_i \geq 0$

For all the $q_i$ such that $q_i > 0$, apply Lagrange multiplier method and calculate the derivatives with respect to $q_i$:

$$\frac{\partial}{\partial q_i}\left(LL - \lambda\left(\sum_i q_i - 1\right)\right) = \frac{f_i \beta}{\alpha p_i + \beta q_i} - \lambda = 0 \tag{8}$$

$$q_i = \frac{f_i}{\lambda} - \frac{\alpha}{\beta} p_i \tag{9}$$

Applying the constraint that $\sum_i q_i = 1$ to all the $q_i > 0$ :

$$\sum_i \left(\frac{f_i}{\lambda} - \frac{\alpha}{\beta} p_i\right) = 1 \tag{10}$$

According to (10), we can get:

$$\lambda = \frac{\sum_{i:q_i>0} f_i}{1 + \frac{\alpha}{\beta}\sum_{i:q_i>0} p_i} \tag{11}$$

Substitute $\lambda$ in equation (9) with the right hand side of equation (11), we get:

$$q_i = \frac{f_i\left(1 + \frac{\alpha}{\beta}\sum_{j:q_j>0} p_j\right)}{\sum_{j:q_j>0} f_j} - \frac{\alpha}{\beta} p_i \tag{12}$$

$$= \frac{\alpha}{\beta} f_i \left(\frac{\frac{\beta}{\alpha} + \sum_{j:q_j>0} p_j}{\sum_{j:q_j>0} f_j} - \frac{p_i}{f_i}\right) \tag{13}$$

Now we can calculate $q_i$ if we know which $q_i$'s are non-zero. The next problem is to find the none-zero $q_i$'s. This is addressed by the following statement.

**Statement**: If $q_i : i = 1 \cdots k$ maximize the objective function (7), $q_2 > 0$ and $\frac{p_1}{f_1} < \frac{p_2}{f_2}$, then $q_1 > 0$.

Proof:

We prove the statement by contradiction.

Let $\Delta q$ represents a small positive number. Suppose $q_1 = 0$, then:

$$\begin{aligned}
& LL(q_1+\Delta q, q_2-\Delta q, q_3, \cdots, q_k) \\
& \quad -LL(q_1, q_2, q_3, \cdots, q_k) && (14)\\
= \; & f_1 \log(\alpha p_1 + \beta(q_1+\Delta q)) \\
& \quad + f_2 \log(\alpha p_2 + \beta(q_2 - \Delta q)) \\
& \quad - f_1 \log(\alpha p_1 + \beta q_1) \\
& \quad - f_2 \log(\alpha p_2 + \beta q_2) && (15)\\
\simeq \; & f_1 \frac{\beta \Delta q}{\alpha p_1 + \beta q_1} - f_2 \frac{\beta \Delta q}{\alpha p_2 + \beta q_2} && (16)\\
= \; & \frac{\beta \Delta q}{\alpha} \left( \frac{f_1}{p_1} - \frac{f_2}{p_2 + \frac{\beta}{\alpha} q_2} \right) && (17)\\
= \; & \frac{\beta \Delta q}{\alpha} \left( \frac{f_1}{p_1} - \frac{\sum_{j:q_j>0} f_j}{\frac{\beta}{\alpha} + \sum_{j:q_j>0} p_j} \right) && (18)\\
> \; & 0 && (19)
\end{aligned}$$

(15) to (16) uses the first order Taylor expansion of log function. (16) to (17) follows from the assumption $q_1 = 0$. (17) to (18) is the result of substituting $q_2$ with Equation (13). (18) to (19) follows from the assumption $\frac{p_1}{f_1} < \frac{p_2}{f_2}$ and the constraint:

$$q_2 = \frac{\alpha}{\beta} f_i \left( \frac{\frac{\beta}{\alpha} + \sum_{j:q_j>0} p_j}{\sum_{j:q_j>0} f_j} - \frac{p_2}{f_2} \right) > 0 \qquad (20)$$

By far, we have shown that:

$$\begin{aligned}
& LL(q_1+\Delta q, q_2-\Delta q, q_3, \cdots, q_k) \\
& \quad > LL(q_1, q_2, q_3, \cdots, q_k) && (21)
\end{aligned}$$

Thus $q_i : i = 1 \cdots k$ do not maximize (7)

Now we finish the proof of the statement.

The direct result of the statement is that all the $q_i$'s greater than 0 correspond to the smallest $\frac{p_i}{f_i}$. So we can use the following algorithm to find the exact multinomial distribution $q$ that maximize the likelihood of observed data:

**Algorithm**: Sort $\frac{p_i}{f_i}$ so that $\frac{f_1}{p_1} > \frac{f_2}{p_2} > \cdots > \frac{f_k}{p_k}$, find $t$ such that

$$\frac{\frac{\beta}{\alpha} + \sum_{j=1}^{t} p_j}{\sum_{j=1}^{t} f_j} - \frac{p_t}{f_t} > 0 \qquad (22)$$

and

$$\frac{\frac{\beta}{\alpha} + \sum_{j=1}^{t+1} p_j}{\sum_{j=1}^{t+1} f_j} - \frac{p_{t+1}}{f_{t+1}} \leq 0 \qquad (23)$$

Then $q_i$'s are given by:

$$q_i = \begin{cases} \frac{f_i}{\lambda} - \frac{\alpha}{\beta} p_i & \text{if } 1 \leq i \leq t \\ 0 & \text{otherwise} \end{cases} \qquad (24)$$

Where $\lambda$ is given by:

$$\lambda = \frac{\sum_{i=1}^{t} f_i}{1 + \frac{\alpha}{\beta} \sum_{i=1}^{t} p_i} \qquad (25)$$

The complexity of the algorithm is the same as sorting, which is $O(k \log(k))$.

## 3. Experimental Results

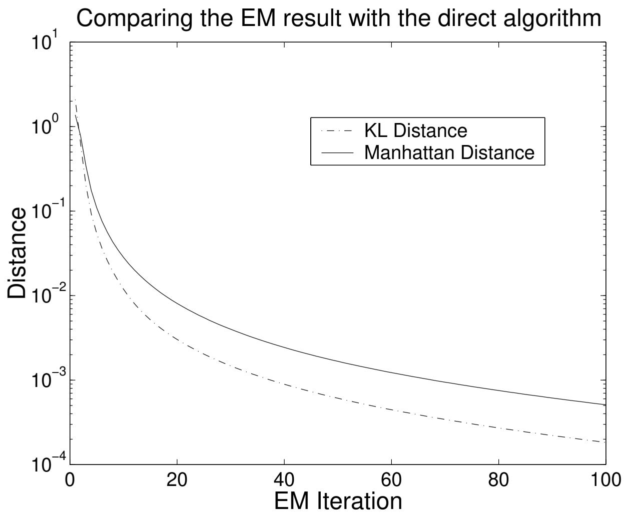


*Figure 1.* The multinomial distribution found by EM algorithm is converging to the distribution calculated directly by our fast algorithm.

We compared our algorithm with EM algorithm in the task of finding language model for a query as describe in (Zhai & Lafferty, 2001) and section 1. In our experiments, we used 20 relevant documents (sampled from AP Wire News and Wall Street Journal dataset from 1988-1990) for a topic as observed training data sequence. $p$ is calculated directly as described in section

1 from 119823 documents in AP Wire News and Wall Street Journal in 1988. There are 2352 unique words in these 20 relevant documents, which means at most 2352 $q_i$'s are none zero, while there are 200542 $p_i$'s are none zero.

Figure 1 shows the Manhattan distance and Kullback-Leibler distance between the multinomial distribution found by EM algorithm at each iteration and the distribution calculated by our algorithm. We can see that the EM result is actually converging to the result calculated directly by our algorithm. As we know, EM is guaranteed to converge to the unique optimal value for mixture of multinomial because the searching surface is convex. Thus empirically, our result is the true optimal.

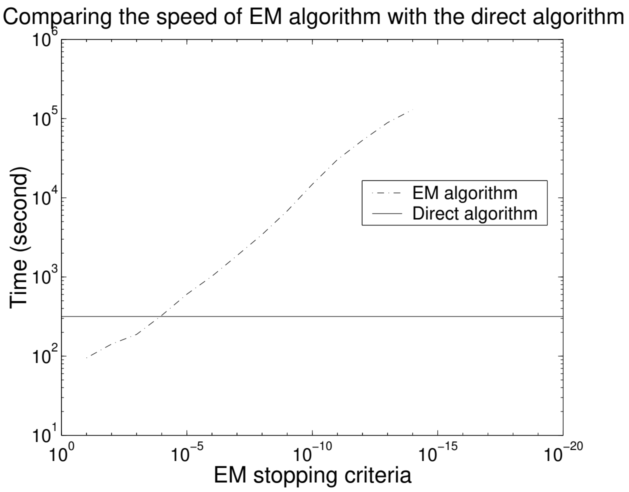


*Figure 2.* The EM speed depends on stopping criteria $10^{-\delta}$, while the new algorithm does not. The vertical axis corresponds to the wall clock time for running the algorithm 50000 times on a PIII 500 PC.

In order to compare the speed of the new algorithm with EM algorithm, we must decide the stopping criteria for EM iterations. One natural way to control the stopping is using the change of log likelihood ($LL$) of data calculated at each iteration (Equation 7). In our experiments, we used the following criteria:

$$\text{Stop if } \textit{change of } LL < 10^{-\delta}$$

Where $\textit{change of } LL = \left| \frac{LL^{(n)} - LL^{(n-1)}}{LL^{(n)}} \right|$, and $LL^{(n)}$ is the log likelihood of the data at the $n$-th iteration.

Figure 2 shows the training time of EM algorithm for different $\delta$. In this experiment, EM speed is similar to the new algorithm when $\delta$ is small. This implies their speed is comparable if the requirement on accuracy is not very high, otherwise the new algorithm is much faster.

## 4. Conclusion

We provide an exact solution and a quick algorithm to solved the problem of find the maximum likelihood estimation of the word mixtures, given fixed mixture weights and the density of another multinomial. This algorithm can be used in the task of ad-hoc information retrieval with relevance feedback, redundancy/novelty detection in information filtering, and other tasks with similar problem settings. Experimental result shows the result of EM algorithm converges to the result calculated exactly with our algorithm, while our algorithm is guaranteed to find the exact unique optimal result at a very fast speed.

## 5. Acknowledgments


We thank Chengxiang Zhai and Thomas Minka on valuable discussions of our algorithm. The work of Yi Zhang and Jamie Callan are supported by Air Force Research Laboratory contract F30602-98-C-0110. Any opinions, findings, conclusions or recommendations expressed in this paper are the authors', and do not necessarily reflect those of the sponsors.


## References


Kraaij, W., Pohlmann, R., & Hiemstra, D. (1999). Twenty-one at trec-8: using language technology for information retrieval. *Proceedings of the Eighth Text REtrieval Conference (TREC-8). National Institute of Standards and Technology, special publication..*

Zhai, C., & Lafferty, J. (2001). Model-based feedback in the language modeling approach to information retrieval. *Proceedings of the Tenth International Conference on Information and Knowledge Management.*

Zhang, Y., Callan, J., & Minka, T. (2002). Novelty and redundancy detection in adaptive filtering. *"Proceedings of 25th Annual International ACM SIGIR Conferenc eon Research and Development in Information Retrieval".*