THE BETA-BINOMIAL MIXTURE MODEL FOR WORD FREQUENCIES IN DOCUMENTS WITH APPLICATIONS TO INFORMATION RETRIEVAL

Stephen A. Lowe
steve@dragonsys.com

Dragon Systems, Inc.
320 Nevada Street
Newton, MA 02460 USA
http://www.dragonsys.com

ABSTRACT
This paper describes a continuous-mixture statistical model for word occurrence frequencies in documents, and the application of that model to the DARPA-sponsored TDT topic identification tasks [1]. This model was originally proposed in 1990 by L. Gillick [2] as a means to account for variation in word frequencies across documents more accurately than the binomial model. The present paper presents further mathematical development of the model, leading to the implementation of a topic-tracking system. Performance results for this system on the Tracking Task in the December 1998 DARPA TDT Evaluation will be shown and compared with Dragon’s existing, more complex multinomial-model-based system. (Results from other systems applied to this task are available in [3].) We will conclude with plans for further development.

1. INTRODUCTION
Previous work at Dragon Systems on topic identification tasks has consistently followed a theme of defining document similarity using statistical measures [2, 4, 5, 6, 7, 8]. To elaborate, for a given document collection we construct a statistical model for the frequencies with which words (or other features, such as bigrams) occur in documents drawn from that collection. For example, we construct a model for the set of documents which are considered to be relevant to a particular topic, or we construct one for the entire space of possible documents (background). The method of construction which we have generally employed is to fit a parametric distribution to a (hopefully) representative sample of documents from the target set. In the cases just mentioned, we would fit parameters for a selection of known on-topic documents to prepare a topic model, or fit parameters for the entire available corpus to produce the background model. Once this is done, decision criteria for assessing relevance of a given document are formulated by using standard statistical tests, usually involving probability ratios.

This approach provides the potential to develop the two central mathematical elements in information retrieval, the term-weighting function and the document similarity measure, from a single theoretical basis. This basis is an optimization problem expressed directly in terms of the likelihood of success of the decision procedure applied to the target task. It leads to a similarity measure which has the appropriate term-weighting function automatically included. The two form a single, unified expression.

The statistical formulation has a long history in the information retrieval field. An often-cited paper on this subject is by Robertson and Spark Jones in 1976 [9], but work more relevant to the line of discussion in the present paper is by Harter in 1975 [10]. However, it appears to the present author that other, ad hoc, term weights and similarity measures are currently in favor in the IR community. If this is so, one is led to ask the question of why an approach with a more complete theoretical underpinning has so far been unable to demonstrate superior performance. One reason that this may have occurred is that the underlying statistical models chosen so far by researchers in this area do not faithfully represent the actual way in which words are distributed across documents. It may be that the ad hoc formulations are, in fact, empirically-derived approximations to the right model (or, at least, a more correct model). Consequently, these formulations produce better results than more theoretically-motivated methods based on the wrong model.

In Dragon’s own work we have generally used the multinomial model in which, for a given source, every word \( w \) is characterized by a generation probability, \( p_w \), and every document from this source is assumed to be created by a sequence of independent random draws from this distribution. (If there is only one word of interest, this model is called the binomial.) Though used effectively for language modeling and even for topic classification, this model may poorly predict the frequency distribution of a word across a collection of documents, as is exhibited in the dash-dot curve on the left of Figure 1. The example shown is fairly typical. This poor fit between data and model may contribute to the need for corrective measures such as excluding from analysis some vocabulary words (stop lists), or applying normalizations that are not motivated by statistical reasoning. The solid curve in Figure 1,
The symbol to be evaluated in closed form:

\[ \int_0^1 dp P_{\text{Bin}}(n | s, p) P_{\text{Beta}}(p | \alpha_w, \beta_w) \]  

(1)

The mixture weight for a given value of \( p \) was chosen to be the well-known Beta density \( P_{\text{Beta}}(p | \alpha_w, \beta_w) \). ([12], pp. 592ff.) It was chosen because it is flexible enough to exhibit a variety of shapes, and yet permits the integral (1) to be evaluated in closed form:

\[ \tilde{P}(n | s, \alpha_w, \beta_w) = \binom{s}{n} \left( \frac{\alpha_w}{\alpha_w + \beta_w} \right)^{n} \left( \frac{1}{\alpha_w + \beta_w} \right)^{s-n} \]  

(2)

The main conceptual points to take from these relations is that the parameter \( \mu \) is the expected value for \( n/s \), and that the variance of the \( P(n | s, \mu, \nu) \) increases with \( \nu \).

### 3. ESTIMATING \( \mu \) AND \( \nu \)

For estimating a model for a given vocabulary word \( w \) from training data, a collection of \( K \) documents consists simply of a vector of document lengths \( s = \{s_1, \ldots, s_K\} \) and a vector of counts for \( w \) in those documents \( n_w = \{n_{w1}, \ldots, n_{wK}\} \). Then we use the Maximum Likelihood Estimate (MLE) to determine preliminary estimates for the parameters \( \mu_w \) and \( \nu_w \):

\[ \left( \hat{\mu}_w, \hat{\nu}_w \right) = \arg\max_{0 \leq \mu \leq 1, \nu \geq 0} \left\{ \log P(n_w | s, \mu, \nu) \right\} \]  

(8)

The probability of the document set can be calculated from the probabilities (1) for the individual documents (using the change of parameters given by equation (4)):

\[ \log P(n_w | s, \mu, \nu) = \sum_{k=1}^{K} \log P(n_{kw} | s_k, \mu, \nu) \]  

(9)

The optimization problem (8) in two dimensions can be solved numerically through fairly standard techniques.
4. APPLICATION TO TRACKING

In the baseline TDT2 Tracking task, a system is presented with a number \( N_t = 4 \) of topic training stories which are known to concern a given target topic. It must then successively examine each of a set of test stories, assign a numerical relevance value, and also issue a “hard” on-topic/off-topic decision. The TDT2 resource data was collected from broadcast and print news sources in the first half of 1998, and is partitioned into Training (Jan.–Feb.), Development Test (Mar.–Apr.), and Evaluation Test (May–June) sets.

A simple system was implemented using a fixed vocabulary \( V \) of size \( N_v = 62 \) words. There was no pre-defined “stop list” of words excluded from the computation. A model \( \mathcal{M}^B = \{ (\mu_{i,w}^B, \nu_{i,w}^B), \ldots, (\mu_{N_v,w}^B, \nu_{N_v,w}^B) \} \) for background was calculated using the set \( \mathcal{D}^B \) of stories designated “NEWS” in the TDT2 Training Set (about 15K stories). The same \( \mathcal{M}^B \) is used for all topics. Then for a topic \( T \), a model \( \mathcal{M}^T \) is constructed from the \( N_t \) topic training stories \( \mathcal{D}^T \). Next, a keyword list \( V^T \) was defined as the subset of \( V \) consisting of words which appear in both the topic and background training sets, and which satisfies:

\[
P(D^T | \mu_{i,w}^T, \nu_{i,w}^T) / P(D^T | \mu_{i,w}^B, \nu_{i,w}^B) > t_{key} \quad (10)
\]

The threshold \( t_{key} \) controls the number of keywords.

A document \( d \) of size \( s^{(d)} \) with word counts \( n_{i,w}^{(d)} \) is evaluated for relevance based on the following score, which is the log product of the topic/background probability ratio computed according to each of the keywords:

\[
S^T(d) = \sum_{w \in V^T} \log \left( \frac{P(n_{i,w}^{(d)} | s^{(d)}, \mu_{i,w}^T, \nu_{i,w}^T)}{P(n_{i,w}^{(d)} | s^{(d)}, \mu_{i,w}^B, \nu_{i,w}^B)} \right) 
\]

(11)

\( S^T \) is something like a log probability ratio, but it is not exactly that, because it does not account for the interdependence of the word counts. For example, \( S^T \) would assign probability to a collection of word counts \( n_{i,w}^{(d)} \) whose sum exceeded the total document length. The system makes an on-topic/off-topic decision by comparing \( S^T(d) \) with a fixed threshold \( t_{on-topic} \). The tendency of the system to favor miss errors over false alarm errors can be adjusted by changing \( t_{on-topic} \).

5. ADJUSTED ESTIMATES FOR \( \mu, \nu \)

The initial experiments were less successful than hoped when compared with other systems. Inspection of the data revealed that there were many words for which \( \nu_w = 0 \). Two common situations were observed in which this occurred: (a) A rare word may happen to occur no more than once in any document; this is observed even in the background data. (b) Since there were only 4 training documents for each topic, any given keyword could easily exhibit low variability within this small subset just by chance. In either of these cases, deviations from the expected value are severely penalized in the score \( S^T(d) \). In case (a), the word appearing twice in the same document is judged too surprising. In case (b), the score is too harsh for a keyword that appears less often than expected.

The problem here is that the estimation procedure only inspects the training data, and has no prior information about the “reasonableness” of different values for \( \nu \). Analysis to determine how best to incorporate this information is underway, but has not been completed as of this writing.

In order to improve the tracking results, some simple adjustments were implemented that increased \( \nu_w \) from its MLE value. The detailed analysis will not be presented, but the basic idea was to correct situation (a) by assuring that \( P(n = 2) \) is not too small, and to fix (b) by assuring that \( P(n = 0) \) is not too small. The expressions used for these adjustments are:

\[
\nu'_{i,w} = \left( \frac{Q}{K_w^+} - \mu_w \right)^2 / \mu_w \quad (12)
\]

\[
\nu''_{i,w} = \lambda_{\min}^2 \mu_w \quad (13)
\]

\[
\nu = \max \{ \nu_w, \nu'_w, \nu''_w \} \quad (14)
\]

\( K_w^+ \) is the number of documents in which \( w \) occurs at least once, and \( Q \) and \( \lambda_{\min}^2 \) are adjustable parameters. Conditions (12), (13) respectively address situations (a), (b).

6. TRACKING RESULTS

The system was tested on the TDT2 Development Test (March–April 1998) data comprising a mixture of text and automatically transcribed broadcast news sources. There are 17 topics in the standard DARPA-defined task. Each topic has 4 training stories. The number of test stories that the system must examine varies from about 4000 to about 17,000 for the different topics, and the number of on-topic targets varies from 1 to 140. A curve mapping out the trade-off between miss vs. false alarm probabilities is created by varying the threshold \( t_{on-topic} \).

In Figure 2, three such curves are presented. Better performance is indicated by a curve which is close to the lower left-hand corner. The lowest (best) curve shows the performance of the final system, in which the MLE estimates \( \mu, \nu \) are adjusted by expressions (12)–(14). The values \( Q = 0.001, \lambda_{\min}^2 = 2, \) and \( \log_2 t_{key} = 2 \) were used. The middle curve shows the same system without those corrections. The third, upper curve is for a system which uses the Binomial model, but which is otherwise very similar to the final system. (See [13] for a more precise comparison.) It is clear that the Beta-Binomial model provides a substantial improvement if the \( \mu, \nu \) parameter values are estimated well, and that the corrections beyond the MLE are important.

Since some system parameters were set based on experience with the Development Test Set, the final system was run on fresh data, the TDT2 Evaluation Test Set. The solid curve in Figure 3 shows the result. The other curve in that
figure is the result from Dragon’s other tracking system, a more complex multinomial-based developed by Yamron [14]. It should be kept in mind that this latter system is much more mature, in that it includes a pre-defined stop list, multiple background models, a smoothing algorithm developed from years of ASR research, filtering of near-topic data from the background, and unsupervised adaptation during the tracking pass. It is encouraging, therefore, that the Beta-Binomial system is able to perform better in part of the experiment space, namely the low-false-alarm regime.

7. CONCLUSION

It is clear from these experiments that the flexibility derived from the Beta-Binomial Mixture Model can be used to obtain superior performance versus a system of equivalent complexity based on the Binomial Model. The current simple implementation even outperforms, in some respects, a more elaborate system based on the multinomial model. Furthermore, the high-false-alarm regime, where the Beta-Binomial did not outperform the multinomial, is precisely the region in which our experiments have demonstrated the greatest potential for improvement if the parameter estimation process is changed. Therefore, we expect that a more principled procedure for incorporating prior expectations into the estimation of $\mu$ and $\nu$ can lead to an overall superior system.

For the future, we note that features other than word counts can be used equally well in this formulation. We would like to try both adjacent and long-range n-grams.

The Beta-Binomial similarity measure can also be used for automatic self-organization of documents into groups (clustering). One problem that needs to be addressed for this task is the large number of parameter estimations that must be performed as documents are tentatively assigned and then reassigned to the emerging clusters. We have reformulated the computation of the sum (9) and its derivatives in a form that will aid this calculation, and will soon be testing it on the TDT2 Detection task.

8. REFERENCES