Estimation of Probabilities from Sparse Data for the Language Model Component of a Speech Recognizer

SLAVA M. KATZ

***Abstract*—The description of a novel type of *m*-gram language model is given. The model offers, via a nonlinear recursive procedure, a computation and space efficient solution to the problem of estimating probabilities from sparse data. This solution compares favorably to other proposed methods. While the method has been developed for and successfully implemented in the IBM Real Time Speech Recognizers, its generality makes it applicable in other areas where the problem of estimating probabilities from sparse data arises.**

Sparseness of data is an inherent property of any real text, and it is a problem that one always encounters while collecting frequency statistics on words and word sequences (m-grams) from a text of finite size. This means that even for a very large data collection, the maximum likelihood estimation method does not allow us to adequately estimate probabilities of rare but nevertheless possible word sequences—many sequences occur only once (''singletons''); many more do not occur at all. Inadequacy of the maximum likelihood estimator and the necessity to estimate the probabilities of m-grams which did not occur in the text constitute the essence of the problem.

The main idea of the proposed solution to the problem is to reduce unreliable probability estimates given by the observed frequencies and redistribute the ''freed'' probability ''mass'' among m-grams which never occurred in the text. The reduction is achieved by replacing maximum likelihood estimates for m-grams having low counts with renormalized Turing's estimates [1], and the redistribution is done via the recursive utilization of lower level conditional distributions. We found Turing's method attractive because of its simplicity and its characterization as the optimal empirical Bayes' estimator of a multinomial probability. Robbins in [2] introduces the empirical Bayes' methodology and Nadas in [3] gives various derivations of the Turing's formula.

Let N be a sample text size and let n_r be the number of words (m-grams) which occurred in the text exactly r times, so that

$$N = \sum_r r n_r . \tag{1}$$

Turing's estimate P_T for a probability of a word (m-gram) which occurred in the sample r times is

$$P_T = \frac{r^*}{N} \tag{2}$$

where

$$r^* = (r + 1) \frac{n_{r+1}}{n_r} . \tag{3}$$

We call a procedure of replacing a count r with a modified count r' ''discounting'' and a ratio r'/r a discount coefficient d_r. When $r' = r^*$, we have Turing's discounting.

Let us denote the m-gram $w_1, \cdots, w_m$ as w_1^m and the number of times it occurred in the sample text as $c(w_1^m)$. Then the maximum likelihood estimate is

$$P_{ML} = \frac{c(w_1^m)}{N} \tag{4}$$

and the Turing's estimate is

$$P_T = \frac{c^*(w_1^m)}{N} \tag{5}$$

where

$$c^*(x) = (c(x) + 1) \frac{n_{c(x)+1}}{n_{c(x)}} . \tag{6}$$

It follows from (1)–(3) that the total probability estimate, using (5), for the set of words (m-gram) that actually occurred in the sample is

$$\sum_{w_1^m : c(w_1^m) > 0} P_T(w_1^m) = 1 - \frac{n_1}{N} . \tag{7}$$

This, in turn, leads to the estimate for the probability of observing some previously unseen m-gram as a fraction n_1/N of ''singletons'' in the text:

$$\sum_{w_1^m : c(w_1^m) = 0} P_T(w_1^m) = \frac{n_1}{N} . \tag{8}$$

On the other hand,

$$\begin{aligned} &\sum_{w_1^m : c(w_1^m) > 0} [P_{ML}(w_1^m) - P_T(w_1^m)] \\ &\quad = \sum_{w_1^m : c(w_1^m) > 0} \delta_{c(w_1^m)} = \sum_{r > 0} n_r (1 - d_r) \frac{r}{N} = \frac{n_1}{N} \end{aligned} \tag{9}$$

where

$$\delta_c = \frac{c}{N} - \frac{c^*}{N} = (1 - d_c) \frac{c}{N} . \tag{10}$$

Thus,

$$\sum_{w_1^m : c(w_1^m) > 0} \delta_{c(w_1^m)} = \sum_{w_1^m : c(w_1^m) = 0} P_T(w_1^m) . \tag{11}$$

Our method is based on the interpretation of δ_c in (11) as a ''contribution'' of an m-gram w_1^m with a count $c(w_1^m)$ to the probability of ''unseen'' m-grams. We go further with this interpretation for the estimating of conditional probabilities $P(w_m | w_1^{m-1})$. Assuming that the $(m-1)$-gram w_1^{m-1} has been observed in the sample text, we introduce an entity $\delta_c^{(\mathrm{cond})}$ analogous to δ_c, given by (10)

$$\delta_{c(w_1^m)}^{(\mathrm{cond})} = (1 - d_{c(w_1^m)}) \frac{c(w_1^m)}{c(w_1^{m-1})} . \tag{12}$$

We now define our estimator $P_s(w_m | w_1^{m-1})$ inductively as follows. Assume that the lower level conditional estimator $P_s(w_m | w_2^{m-1})$ has been defined. Then, when $c(w_1^{m-1}) > 0$,

$$P_s(w_m | w_1^{m-1}) = \tilde{P}(w_m | w_1^{m-1}) = d_{c(w_1^m)} \frac{c(w_1^m)}{c(w_1^{m-1})} \tag{13}$$

gives the conditional probability estimate for words w_m observed in the sample text after $w_1^{m-1} (c(w_1^m) > 0)$. It is convenient to define a function $\tilde{\beta}$ by

$$\begin{aligned} \tilde{\beta}(w_1^{m-1}) &= \sum_{w_m : c(w_1^m) > 0} \delta_{c(w_1^m)}^{(\mathrm{cond})} \\ &= 1 - \sum_{w_m : c(w_1^m) > 0} \tilde{P}(w_m | w_1^{m-1}) . \end{aligned} \tag{14}$$

This gives an estimate of the sum of conditional probabilities of all words w_m which never followed $w_1^{m-1} (c(w_1^m) = 0)$. We distribute the probability ''mass'' $\tilde{\beta}$, defined by (14), among w_m for which $c(w_1^m) = 0$ using a previously defined (by induction) lower level conditional distribution $P_s(w_m | w_2^{m-1})$:

$$P_s(w_m | w_1^{m-1}) = \alpha P_s(w_m | w_2^{m-1}) \tag{15}$$

Manuscript received May 1, 1986; revised September 2, 1986.
The author is with the Continuous Speech Recognition Group, Department of Computer Sciences, IBM T. J. Watson Research Center, Yorktown Heights, NY 10598.
IEEE Log Number 8611734.

0096-3518/87/0300-0400$01.00 © 1987 IEEE

where

$$\alpha = \alpha(w_1^{m-1}) = \frac{\tilde{\beta}(w_1^{m-1})}{\sum_{w_m : c(w_1^m)=0} P_s(w_m | w_2^{m-1})} = \frac{1 - \sum_{w_m : c(w_1^m)>0} \tilde{P}(w_m | w_1^{m-1})}{1 - \sum_{w_m : c(w_1^m)>0} \tilde{P}(w_m | w_2^{m-1})} \quad (16)$$

is a normalizing constant. When $c(w_1^{m-1}) = 0$, then we define

$$P_s(w_m | w_1^{m-1}) = P_s(w_m | w_2^{m-1}). \quad (17)$$

Complementing $\tilde{P}$ and $\tilde{\beta}$ definitions, given by (13) and (14), for the case when $c(w_1^{m-1}) = 0$, with

$$\tilde{P}(w_m | w_1^{m-1}) = 0$$

and

$$\tilde{\beta}(w_1^{m-1}) = 1,$$

we finally combine estimates (13), (15), and (17) in the following recursive expression for the conditional probability distribution:

$$P_s(w_m | w_1^{m-1}) = \tilde{P}(w_m | w_1^{m-1}) + \theta(\tilde{P}(w_m | w_1^{m-1})) \cdot \alpha(w_1^{m-1}) \, P_s(w_m | w_2^{m-1}) \quad (18)$$

where

$$\theta(x) = \begin{cases} 1, & \text{if } x = 0 \\ 0, & \text{otherwise.} \end{cases} \quad (19)$$

We now propose a somewhat modified version of the distribution given by (18). We shall leave intact the estimate n_1/N for the probability of all unseen m-grams and we shall not discount high values of counts $c > k$, considering them as reliable. To achieve this, we redefine

$$d_r = 1, \qquad \text{for } r > k \quad (20)$$

and we shall correspondingly adjust Turing's original discount coefficients d_r for $r \leq k$ so that the equation expressing the balance between the "contributions" and the probability of unseen m-grams

$$\sum_{w_m : c(w_1^m) > 0} \delta_{c(w_1^m)} = \sum_{1 \leq r \leq k} n_r (1 - d_r) \frac{r}{N} = \frac{n_1}{N} \quad (21)$$

is satisfied. Equation (21) is analogous to (9). We obtain an adjustment of the d_r looking for a solution of (21) in the form

$$(1 - d_r) = \mu \left(1 - \frac{r^*}{r} \right) \quad (22)$$

where μ is a constant. The unique solution is given by

$$d_r = \frac{\dfrac{r^*}{r} - \dfrac{(k+1)\, n_{k+1}}{n_1}}{1 - \dfrac{(k+1)\, n_{k+1}}{n_1}}, \qquad \text{for } 1 \leq r \leq k. \quad (23)$$

Equality (22) is equivalent to the requirement that newly defined count "contributions" $(r/N - r'/N)$ are proportional to the Turing's "contributions" $(r/N - r^*/N)$. As for the value for the parameter k, in practice, $k = 5$ or so is a good choice. Other ways of computing discount coefficients could be suggested for, in general, the model is not very sensitive to the actual values of d_r. The latter holds as long as an estimate for the probability of observing some previously unseen m-gram is reasonable. When data are very sparse, an estimate n_1/N is well justified—there is not too much of a difference in seeing an m-gram once or not seeing it at all.

Numerical values for α's can be precomputed, and that provides the method's computational efficiency. A 3-gram model constructed in accordance with formulas (18), (16), and (23) was implemented as a language model component of the Real-Time Speech Recognizer [4]. On the other hand, as our experiments show, setting $d_1 = 0$, which is equivalent to discarding all "singletons," does not affect the model performance, and thus provides substantial saving in space needed for the language model. We took advantage of it in constructing a compact language model for the PC-based Speech Recognizer [5].

The approach described compares favorably to other proposed methods. Table I gives the perplexity computation results for three models being compared. The first column gives the results for the deleted estimation method by Jelinek and Mercer [6], the second column for the parametric empirical Bayes' method by Nádas [7], and the third one for our "back off" estimation method.

TABLE I
PERPLEXITY RESULTS ON 100 TEST SENTENCES

Model	1	2	3
2-gram	118	119	117
3-gram	89	91	88

Perplexity, defined as 2^H where

$$H = -\frac{1}{L - m + 1} \sum_{l=m}^{l=L} \log_2 P(w_l | w_{l-m+1}^{l-1})$$

$w_1, \cdots, w_L$ is the test word sequence and $m = 2, 3$ for a 2-gram and 3-gram model, respectively, is used here to characterize the performance of the model—the lower the perplexity, the better the model. The statistics used for constructing the models were obtained from approximately 750 000 words of an office correspondence database; 100 sentences were used for testing. This and other experiments showed that our method consistently yields perplexity results which are at least as good as those obtained by other methods. Meanwhile, our model is much easier to construct, implement, and use. In closing, we wish to emphasize that the novelty of our approach lies in the nonlinear "back-off" procedure which utilizes an explicit estimation of the probability of unseen m-grams and not in the details of computation such as the use of Turing's formulas.

REFERENCES

[1] I. J. Good, "The population frequencies of species and the estimation of population parameters," *Biometrika*, vol. 40, no. 3 and 4, pp. 237–264, 1953.

[2] H. E. Robbins, "An empirical Bayes approach to statistics," in *Proc. 3rd Berkeley Symp. Math. Statist. Prob.*, vol. 1, 1956, pp. 157–164.

[3] A. Nádas, "On Turing's formula for word probabilities," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, pp. 1414–1416, Dec. 1985.

[4] A. Averbuch *et al.*, "A real-time isolated-word, speech recognition system for dictation transcription," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, 1985, pp. 858–861.

[5] A. Averbuch *et al.*, "An IBM PC based large-vocabulary isolated-utterance speech recognizer," in *Proc. IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tokyo, Japan, Apr. 1986.

[6] F. Jelinek and R. L. Mercer, "Interpolated estimation of Markov source parameters from sparse data," in *Pattern Recognition in Practice*, E. S. Gelsema and L. N. Kanal, Eds. Amsterdam: North-Holland, 1980.

[7] A. Nádas, "Estimation of probabilities in the language model of the IBM Speech Recognition System," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, pp. 859–861, Aug. 1984.