MINIMUM VERIFICATION ERROR TRAINING FOR TOPIC VERIFICATION

Hong-Kwang Jeff Kuo[*], *Chin-Hui Lee*[†], *Imed Zitouni, Eric Fosler-Lussier*[‡]

Bell Labs, Lucent Technologies, 600 Mountain Ave., Murray Hill, NJ 07974-0636, U.S.A.
hkuo@us.ibm.com, chin.lee@ece.gatech.edu,
zitouni@research.bell-labs.com, fosler@ee.columbia.edu

ABSTRACT

In this paper we propose a new formulation of minimum verification error training and apply it to the problem of topic verification as an example. In topic verification, a decision is made as to whether a document truly belongs to a particular topic of interest. Such a decision typically depends on a comparison between a model for the desired topic and a model for background topics, using a decision threshold. We propose modeling the background topics as a cohort model consisting of a weighted combination of the M closest topics discovered from the training data. The weights and the decision threshold are optimized using the generalized probabilistic descent algorithm to explicitly minimize the verification error rate, which is defined to be a weighted sum of the Type I (false rejection) and Type II (false acceptance) errors.

1. INTRODUCTION

In a natural language call routing application, callers give a description of what they want and are automatically routed to the right department (or directed to a human operator when the system is unable to determine the caller's intent with certainty). In probabilistic approaches to this task, call routing is treated as an instance of document classification, where a collection of labeled documents is used for training and the task is to determine the class of a test document. Each destination in the call center is thus treated as a collection of documents (transcriptions of calls routed to that destination), and a new caller request is evaluated in terms of relevance to each destination [1]. Such a framework is also applicable to topic identification or topic spotting.

In this paper, our focus is not on classifying call documents, but rather the related problem of verifying whether a document truly belongs to a particular topic of interest. This is similar in spirit to other verification paradigms, such as utterance verification or speaker verification [9, 10, 6].

Discriminative training has been found to be very effective for training classifiers used for natural language call routing [3, 4]. The algorithm not only reduces the classification error rate significantly but also provides other benefits, including portability and increased score separation of the correct class from competing classes. We showed recently that the classifier performance can be improved dramatically with little manual tuning for a topic identification task on the Switchboard corpus [5].

Although discriminative training has been shown to be effective for training classifiers, the minimum classification error criterion used in training is not exactly the desired criterion of minimum verification error (MVE). We describe here an effort to achieve the MVE criterion by adjusting parameters in the verification model using a generalized probabilistic descent (GPD) algorithm [2]. We have seen in the past for discriminative methods that improvements in the training set often carry over to the test set, so we expect that applying this method to verification will also improve performance.

In the next section, we introduce the concept of minimum verification error and describe an algorithm to minimize this metric. We then present the results of initial experiments on the Switchboard task with 66 topics, illustrating the effects of minimum verification error training.

2. MINIMUM VERIFICATION ERROR

Let E_{k1} = total Type I (false rejection) error and E_{k2} = total Type II (false acceptance) error for class k. We would like to minimize the class-specific total weighted verification error:

$$E_{kw} = w_{k1} E_{k1} + w_{k2} E_{k2}, \tag{1}$$

where w_{k1} and w_{k2} are constants set by the constraints of the particular application. For example, in certain applications, a false rejection (missing an important document) may incur a larger cost than a false acceptance (which can be rejected by humans). In this case, w_{k1} will be set to be much larger than w_{k2}.

Given that we want to minimize the total weighted error, we now lay out a GPD formulation for minimizing this metric with respect to two types of parameters: the threshold and the weights for members of a competing cohort set used for verification. Note that we will not be adjusting any of the parameters of the classifier itself, which would already have been trained using the minimum classification error (MCE) criterion [4].

Given a document or user request $\vec{x}_i$, a destination or class k, we define the *misverification function* as

$$d_k(\vec{x}, \alpha, \theta) = -g_k(\vec{x}) + G_k(\vec{x}, \alpha) - \theta_k, \tag{2}$$

[*] This work was done at Bell Labs, Lucent. H.-K. J. Kuo is currently at IBM T. J. Watson Research Center, 1101 Kitchawan Road, Route 134, Yorktown Heights, NY 10598, USA.

[†] C.-H. Lee is currently at Georgia Institute of Technology, School of Electrical and Computer Engineering, Atlanta, GA 30332, USA.

[‡] E. Fosler-Lussier is currently at Columbia University, New York, NY, USA.

where $g_k(\vec{x})$ is the discriminant function defined to be a cosine similarity metric [3] and

$$G_k(\vec{x}, \alpha) = \left[\frac{1}{M} \sum_{j=1}^{M} \alpha_{kj} g_j(\vec{x})^{\eta} \right]^{\frac{1}{\eta}} \tag{3}$$

is the *anti-discriminant function* of the input $\vec{x}_i$ in class k, M is the number of top competing models representing the cohort, α_{kj} is the weight of the jth cohort member that competes with class k, and θ_k is the threshold for performing verification. Note that in the limit as the positive parameter $\eta \to \infty$, the anti-discriminant function is dominated by the biggest competing discriminant function: $G_k(\vec{x}, R) \to \max_{j \neq k} g_j(\vec{x}, R)$, where the α terms drop out of the equation. This makes sense because in the limit, the discriminant should be compared with the best competitor, not a weighted version.

If $\vec{x}_i$ belongs to class k, and $d_k > 0$, this is a false rejection error. If $\vec{x}_i$ does not belong to class k, and $d_k < 0$, this is a false acceptance error. Smoothed representations of these two types of errors are created using a smooth differentiable 0-1 function such as the sigmoid function l_k:

$$E_{k1} = \frac{1}{N_{k1}} \sum_{x_i \in C_k} l_k(d_k(\vec{x}_i, \alpha, \theta)), \tag{4}$$

$$E_{k2} = \frac{1}{N_{k2}} \sum_{x_i \notin C_k} l_k(-d_k(\vec{x}_i, \alpha, \theta)), \tag{5}$$

where C_k represents class k, and N_{k1} and N_{k2} are the number of training samples that are in C_k and not in C_k, respectively. More specifically, the first sigmoid function used for false rejection errors in Equation 4 is

$$l_k(d_k(\vec{x})) = \frac{1}{1 + \exp(-\gamma d_k)}, \tag{6}$$

where γ is a constant which controls the slope of the sigmoid function. The second sigmoid function used for false acceptance errors in Equation 5 is a reversed version, defined as:

$$l_k(-d_k(\vec{x})) = \frac{1}{1 + \exp(\gamma d_k)}. \tag{7}$$

The class-specific empirical loss for class k, for the entire training set consisting of N training vectors is then given by:

$$\begin{aligned} L_k(\alpha, \theta) &= w_{k1} E_{k1} + w_{k2} E_{k2} \\ &= \tfrac{w_{k1}}{N_{k1}} \textstyle\sum_{\vec{x}_i \in C_k} l_k(d_k(\vec{x}_i, \alpha, \theta) \\ &\quad + \tfrac{w_{k2}}{N_{k2}} \textstyle\sum_{\vec{x}_f \notin C_k} l_k(-d_k(\vec{x}_f, \alpha, \theta). \end{aligned} \tag{8}$$

Note that the empirical loss is essentially a smoothed function approximating the total weighted error rate that is differentiable so that it can be used in gradient descent optimization.

We will be minimizing the total weighted error with respect to the cohort weights α_{kj} and the verification threshold θ_k. Let V be a vector of these parameters over which we are trying to optimize. Using the GPD algorithm, we would iteratively optimize these parameters to reduce the total weighted verification error:

$$V_{t+1} = V_t - \epsilon_t \nabla L_k(V), \tag{9}$$

where $\nabla L_k(V)$ contains components of $\frac{\partial L}{\partial \alpha_{kj}}$ and $\frac{\partial L}{\partial \theta_k}$.

These parameters are iteratively adjusted to minimize the empirical loss in Equation 8. The update equations for θ_k works out to be quite simple:

$$\theta_{k,t+1} = \theta_{k,t} + \epsilon_{t,\theta} \left(\frac{w_{k1}}{N_{k1}} \sum_{\vec{x}_i \in C_k} \frac{\partial l_k}{\partial d_k} - \frac{w_{k2}}{N_{k2}} \sum_{\vec{x}_f \notin C_k} \frac{\partial l_k}{\partial d_k} \right). \tag{10}$$

Intuitively, for each class, we examine each training sample. If the training sample falls within the decision boundary (region where the slope of the sigmoid is relatively large), the threshold θ_k is adjusted as follows. If the sample is from the correct class, the threshold is adjusted to the right (more positive) by an amount weighted by w_{k1} in order to reduce the Type I error; otherwise if it is not from the correct class, the threshold is adjusted to the left (more negative) by an amount weighted by w_{k2} to reduce the Type II error.

The update equations for the α parameters are as follows:

$$\alpha_{kj,t+1} = \alpha_{kj,t} - \epsilon_{t,\alpha} \frac{\partial E_w}{\partial \alpha_{kj}}, \tag{11}$$

where $\frac{\partial E_w}{\partial \alpha_{kj}}$ equals:

$$\frac{1}{\eta M} \left[\frac{w_{k1}}{N_{k1}} \sum_{\vec{x}_i \in C_k} \frac{\partial l_k}{\partial d_k} G_k \left(\frac{g_j}{G_k} \right)^{\eta} - \frac{w_{k2}}{N_{k2}} \sum_{\vec{x}_f \notin C_k} \frac{\partial l_k}{\partial d_k} G_k \left(\frac{g_j}{G_k} \right)^{\eta} \right]. \tag{12}$$

Intuitively, the cohort weights α_{kj} are adjusted by an amount proportional to the slope of the sigmoid, w_{k1} or w_{k2}, and the magnitude of the anti-discriminant function G_k. Also, the adjustment depends on $(g_j/G_k)^{\eta}$, where g_j is the discriminant function of class j, one of the competitors included in G_k. That is, the adjustment for α_{kJ} depends on how important class J is. If η is large, only the α_{kJ} associated with the best competing model J will be adjusted. Notice also the sign of the adjustment for training samples that are in or not in C_k. For the ones in C_k, the adjustment to α_{kj} is negative; this will tend to decrease G_k and d_k, and therefore reduce Type I error. Likewise, for the training samples not in C_k, the adjustment to α_{kj} is positive; this will tend to increase G_k and d_k, and thereby reduce Type II error.

3. EXPERIMENTAL SETUP

Training and test data were taken from the text transcripts of the Switchboard database [8]. Each conversation side was treated as a separate document; similar to [7] we also removed the first 15 seconds of a conversation because the speakers often said “ok, we’re supposed to talk about X” at the start of a conversation. A set of 66 topics were identified within the corpus.[1] The data were divided into half for training and testing, with the test set drawn so that the distribution of test topics roughly matched that of the training set.

[1] We also prepared a subset of 10 topics, corresponding to the topics in [7], but the results are not reported here.

	baseline	after DT
Training Set	27.9%	2.4%
Test Set	44.6%	13.5%

Table 1. Classification error rate before and after DT

4. RESULTS

First we report some topic classification results for the 66 topics as a prelude to the topic verification experiments. Table 1 shows the classification error rate of the baseline classifier trained using maximum likelihood (ML) (counting based) and after discriminative training using the minimum classification error (MCE) criterion. Note that these results are different from the ones reported previously [5] because of the differences in data preparation, e.g. only one side of conversation is used, the first few seconds are removed, and less data are used for training to reserve more data for the test set. The basic algorithms are the same.

In our topic verification experiments, we arbitrarily chose w_{k1} = w_{k2} = 0.5, keeping in mind that these two values will be dictated by the needs of a particular application and are fixed values. The background model was represented by a cohort model consisting of $M = 30$ models of individual classes. The initial values of α_{kj} for these 30 classes were set to 1, with the other α_{kj} values set to 0. The threshold θ_k was chosen to be 0 initially.

We had chosen to define α_{kj} to be the weights of g_j^η and not of g_j. The motivation was that as $\eta \to \infty$, we wanted $G_k(\vec{x}) \to \max_{j \neq k} g_j(\vec{x})$, rather than $\alpha_{kj} \max_{j \neq k} g_j(\vec{x})$. However, in this case, we have to be careful not to set η to be too large a value; otherwise, G_k will be dominated by the most prominent term, and the α_{kj} terms will have no effect. Consequently, we use a small value of $\eta = 2$; this can be thought of as a way of compensating for unknown probability densities of the background model by using a smoothed combination of competing models.

Initially we had chosen to use the same ϵ_t for training both the α_{kj} and θ_k parameters for each class. However, we discovered that the adjustments made to the α_{kj} parameters were much smaller than those for θ_k. In the preliminary results we report below, we used a different learning step size for the two types of parameters.

Many of the topic classes had very low verification error rate. We therefore chose a class with one of the highest verification error rates to illustrate the effects of MCE and MVE training.

Using the maximum likelihood (ML) classifier for verification, we obtained a weighted verification error of 20% for the training data and 41% for the test data. Figure 1 shows the distribution of the misverification function for the training data. The impostor distribution (*out-of-class* tokens) is shown using a dashed line, while the true distribution (*in-class* tokens) is shown as a histogram. The optimal threshold was found by balancing the Type I and II errors according to the weights and is shown as a dotted line at 0 in the figure. Type I (false rejection) and Type II (false acceptance) errors are also highlighted using dark gray and black shading, respectively. Figure 1 clearly shows the high degree of overlap between the true and impostor distributions using the ML classifier.

Using the MCE trained classifier for verification, we obtained a weighted verification error of 6.4% and 11.8%, for the training and test data, respectively, representing a large improvement over the ML classifier. Figure 2 shows the distribution of the misverification function for the training data. It is apparent that the separation between the true and impostor distributions has been increased, resulting in a much lower verification error.

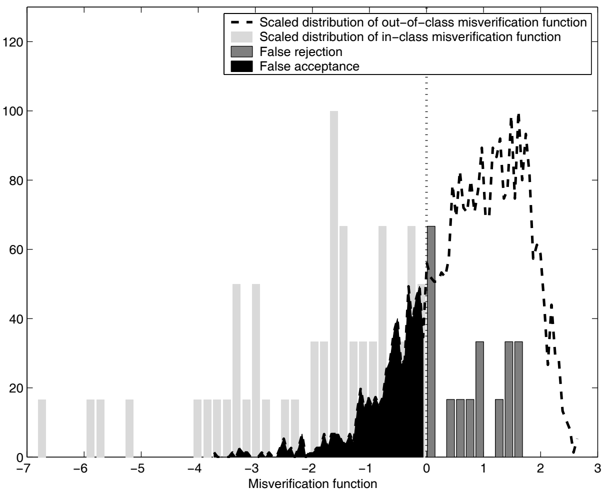

Fig. 1. Distribution of the misverification function using the ML classifier.

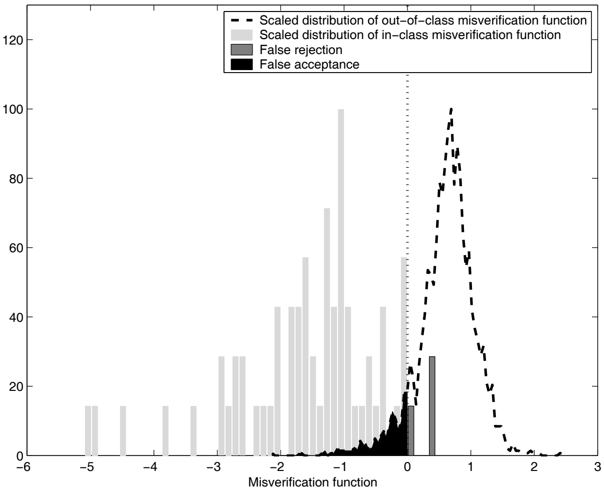

Fig. 2. Distribution of the misverification function for the MCE classifier before MVE training.

Next we perform MVE training using the MCE classifier. Recall that the baseline results shown in Figure 2 are based on $\alpha_{kj} = 1$ for the top M classes. The following set of GPD parameters were used: $\eta = 2$, $\gamma = 4$, $\epsilon_0 = 2$. ϵ_t was reduced every 3 iterations, and a total of 20 iterations were run. θ_k was set to be the maximum d_k value for the true tokens as an initial starting point. After MVE training, the verification error was reduced to 3.8% for the training data and to 8.8% for the test data. Table 2 shows a summary of the improvements in the total weighted verification error rate over the original ML classifier for the MCE classifier and the MCE classifier plus MVE training.

Figure 3 shows the distribution of the misverification function after MVE training for the training data. The shapes of the distributions seem to have changed, perhaps representing a better weighting of the cohort models of the background model. The Type II (false positive) error rate has been reduced without any increase in the Type I (false rejection) error rate.

	Weighted Verification Error	
	Training Set	Test Set
ML Classifier	20%	41%
MCE Classifier	6.4%	11.8%
MCE + MVE	3.8%	8.8%

Table 2. Total weighted verification error rate is reduced using MCE and MVE training.

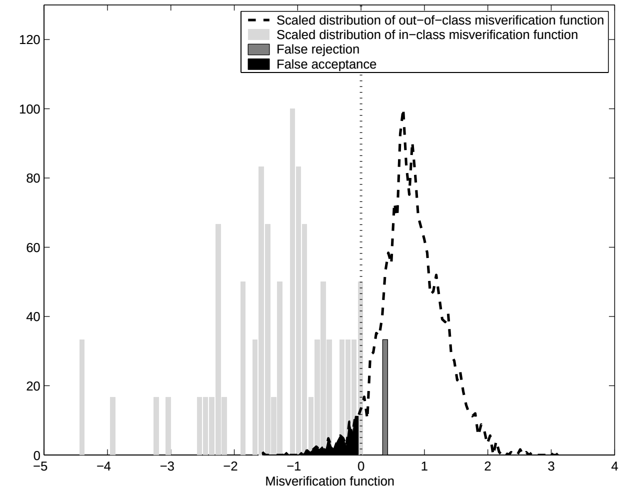

Fig. 3. Distribution of the misverification function after MVE training.

5. DISCUSSION AND CONCLUSIONS

In this paper, we proposed a new formulation of minimum verification error training. The goal is to minimize an error metric that is based on a weighted sum of Type I and Type II errors. The weights are assumed to be pre-determined by the application requirements.

The background model is formulated as a weighted combination of close competing classes (cohort). The particular weights for this combination, as well as the decision threshold used for verification are the parameters that are optimized to achieve minimum verification error. This is achieved using a generalized probabilistic descent algorithm that iteratively reduces a smoothed function of the verification error. Preliminary results show promising improvements based on this simple MVE formulation. More research has to be done to investigate how best to improve the separation between the in-class samples and the out-of-class samples to reduce the verification error. In particular, instead of using MCE models, the MVE criterion can be used explicitly to re-train the model parameters for each topic class.

We highlight a few differences between the current MVE training and the MCE training that we had proposed previously [4] for topic identification in call routing. MVE is tuning for individual classes separately to minimize the verification error whereas MCE is a global adjustment for all training samples to minimize the classification error. Since verification is a two class problem, it is in some ways an easier problem. However, the background model is always difficult to model since we cannot usually anticipate all the possible impostors; in this paper, we use a cohort model to try to address this unsolved problem.

In MVE training, parameters are adjusted according to the cost corresponding to Type I and II errors. Although we had chosen to use uniform weights of 0.5 as a simple example to illustrate the usefulness of MVE, it is straightforward to vary the weights for Type I and II errors to generate results for different operating points on an ROC curve. Finally, although this paper focused on topic verification, MVE training may also be useful for other verification problems such as speaker verification or utterance verification (ASR confidence measures).

6. REFERENCES

[1] J. Chu-Carroll and B. Carpenter, "Vector-based natural language call routing," *Computational Linguistics*, vol. 25, no. 3, pp. 361–388, 1999.

[2] S. Katagiri, B.-H. Juang, and C.-H. Lee, "Pattern recognition using a family of design algorithms based upon the generalized probabilistic descent method," *Proceedings of the IEEE*, vol. 86, pp. 2345–2373, Nov. 1998.

[3] H.-K. J. Kuo and C.-H. Lee, "Discriminative training in natural language call routing," in *Proc. ICSLP-2000*, (Beijing, China), Oct. 2000.

[4] H.-K. J. Kuo and C.-H. Lee, "Discriminative training of natural language call routers," *Accepted to IEEE Transactions on Speech and Audio Processing*, 2003.

[5] H.-K. J. Kuo, C.-H. Lee, I. Zitouni, and E. Fosler-Lussier, "Discriminative training for call classification and routing," in *ICSLP'2002*, (Denver, CO), 2002.

[6] C.-H. Lee, "A tutorial on speaker and speech verification," in *Proc. NORSIG-98*, (Vigso, Denmark), pp. 9–16, June 1998.

[7] J. McDonough, K. Ng, P. Jeanrenaud, H. Gish, and J. R. Rohlicek, "Approaches to topic identification on the switchboard corpus," in *ICASSP'94*, (Adelaide, Australia), 1994.

[8] NIST, "Switchboard corpus: Recorded telephone conversations." National Institute of Standards and Technology Speech Disc 9-1 to 9-25, October 1992.

[9] M. G. Rahim, C.-H. Lee, and B.-H. Juang, "Discriminative utterance verification for connected digits recognition," *IEEE Trans. on Speech and Audio Processing*, vol. 5, pp. 266–277, May 1997.

[10] R. Sukkar, A. R. Setlur, C.-H. Lee, and J. Jacob, "Verifying and correcting string hypotheses using discriminative utterance verification," *Speech Communication*, vol. 22, pp. 333–342, 1997.