MAXIMUM ENTROPY MODELS FOR SPEECH CONFIDENCE ESTIMATION

Claudio Estienne[1]*, Alberto Sanchis*[2]*, Alfons Juan*[2] *and Enrique Vidal*[2]

[1]Facultad de Ingeniería, Universidad de Buenos Aires, Argentina.
[2]Departament de Sistemes Informàtics i Computació. Universitat Politècnica de València, Spain.
cestien@fi.uba.ar, josanna@dsic.upv.es, ajuan@dsic.upv.es, evidal@dsic.upv.es

ABSTRACT

In this work we implement a confidence estimation system based on a Naive Bayes classifier, by using the maximum entropy paradigm. The model takes information from various sources including a set of scores which have proved to be useful in confidence estimation tasks. Two different approaches are modeled. First a basic model which takes advantages of smoothing techniques used in a previous work, and second an optimized model, which is designed to hold a set of very few but essential characteristics of the model, without decrease in the performance. A considerably reduction in the number of parameters is obtained compared to the basic model. Both models are evaluated with two different corpora and compared to a model previously developed.

Index Terms— confidence estimation, maximum entropy, confidence measures, speech recognition.

1. INTRODUCTION

Confidence estimation has been extensively studied for speech recognition [1, 2]. Its basic goal is to estimate a confidence measure for each word in a given hypothesis, in order to locate those words, if any, that are likely to be incorrectly recognized. It can be seen as a two-class pa ttern recognition problem in which each hypothesized word is transformed into a vector and then classified as either correct or incorrect. This view provides a solid, well-known framework within which accurate dichotomizers (two-class classifiers) can be derived. In advance, we will denote these features as *scores* in order to distinguish them from the features defined in the maximum entropy models.

The maximum entropy principle has been successfully applied in many speech processing areas, including language models [8, 10, 11] and natural language processing [9]. It is a well known method to join information captured from various knowledge sources. In this work we will use the maximum entropy paradigm to model likelihood distributions of a classifier used in confidence estimation.

The rest of the paper is divided as follow, in section 2 we present the confidence estimation paradigm. In section 3 we describe the maximum entropy approximation to the confidence estimation problem. In section 4 we present some experiments performed in two different data bases. Finally, in sections 5 and 6, we discuss obtained results and give some concluding remarks.

Work partially supported by the Spanish research programme Consolider Ingenio 2010: MIPRCV (CSD2007-00018) and the EC (FEDER) and the Spanish MEC under grant TIN2006-15694-CO2-01.

2. CONFIDENCE ESTIMATION

Confidence estimation can be seen as a two-class pattern recognition problem in which each hypothesized word is transformed into a vector of *scores* and then classified as either correct or incorrect. The basic problem then is to decide which predictor (pattern) *scores* and classification model should be used.

2.1. Predictor Scores

Different kind of *scores* have been used for confidence estimation in speech recognition [1, 3, 4]. In this work, we have selected a set of well-known *scores* that have proved to be very useful.

2.1.1. Posterior probabilities computed on word graphs

A word graph G is a directed, acyclic, weighted graph. The nodes corresponds to discrete points in time. The edges are triplets $[w,\ s,\ e]$, where w is the hypothesized word from node s to node e. The weights are the recognition scores associated to the word graph edges. Any path from the initial to the final node forms a hypothesis h. Given the acoustic observations $\mathbf{\Theta}_1^T$, the posterior probability for a specific word (edge) $[w,\ s,\ e]$ can be computed by summing up the posterior probabilities of all hypotheses of the word graph containing the edge $[w,\ s,\ e]$:

$$P([w,\ s,\ e] \mid \mathbf{\Theta}_1^T) = \frac{1}{P(\mathbf{\Theta}_1^T)} \sum_{\substack{h \in G : \\ \exists [w', s', e'] : \\ w' = w, s' = s, e' = e}} P(h,\ \mathbf{\Theta}_1^T) \tag{1}$$

The probability of the sequence of acoustic observations $P(\mathbf{\Theta}_1^T)$ can be computed by summing up the posterior probabilities of all word graph hypotheses:

$$P(\mathbf{\Theta}_1^T) = \sum_h P(h\,, \mathbf{\Theta}_1^T) \tag{2}$$

The silence arcs are considered in the same manner as the word edges. There is not a special treatment for this kind of arcs since they can be considered as a part of the hypotheses. These posterior probabilities can be efficiently computed based on the well-known *forward-backward* algorithm [1].The posterior probability defined in (1) doesn't perform well because of a word w can occur with slightly different starting and ending times. This effect is represented in the word graph by different word graph edges and the posterior probability mass of the word is splitted among the different word segmentation [1].

To circumvent this problem, we have considered two methods following the ideas proposed in [1]. Given a specific word (edge)

$[w,\ s,\ e]$ and a specific point in time $t \in (s, e)$, we compute the posterior probability of the word w at time t by summing up the posterior probabilities of the word graph edges $[w, s', e']$ with identical word w and for which t is within the interval time (s', e'):

$$P_t([w,\ s,\ e] \mid \mathbf{\Theta}_1^T) = \sum_{\forall [s',e]:\exists [w',s',e'],t\in(s',e')} P([w,\ s',\ e'] \mid \mathbf{\Theta}_1^T) \tag{3}$$

Based on (3), two different variants of the posterior probabilities are computed for a specific word $[w,\ s,\ e]$:
The median of the frame time posterior probabilities (PostMed) and the maximum (PostMax):

$$P([w,\ s,\ e] \mid \mathbf{\Theta}_1^T) = \frac{1}{e-s+1}\sum_{t=s}^{e} P_t([w,\ s,\ e] \mid \mathbf{\Theta}_1^T) \tag{4}$$

$$P([w,\ s,\ e] \mid \mathbf{\Theta}_1^T) = \max_{s\le t\le e} P_t([w,\ s,\ e] \mid \mathbf{\Theta}_1^T) \tag{5}$$

2.1.2. Alternative predictor scores

Acoustic stability (AS): Number of times that a hypothesized word appears at the same position (as computed by Levenshtein alignment) in K alternative outputs of the speech recognizer obtained using different values of the *Grammar Scale Factor* (GSF), i.e. a weighting between acoustic and language model scores.
AcScore (AcSc): The acoustic log-score of the word divided by its number of phones.
Word Trellis Stability (WTS): This feature was proposed in [3]. Given a word w and its starting and ending times $[s,\ e]$, two variants of the WTS are computed as:

$$\begin{aligned} WTS(w) &= \max_{s\le t'\le e} \frac{C(w,t')}{\sum_{w'} C(w',t')} \\ C(w,t') &= \sum_{t=t'}^{T} \sum_{h\in\mathcal{H}_t(w,t')} (\alpha_f - \alpha_i) \end{aligned}$$

where T is the number of frames of the given utterance, $\mathcal{H}_t$ is a set of word-boundary partial hypotheses that are most probable at time t for a certain range of GSF values $[\alpha_i,\alpha_f]$. In addition, in each hypothesis of $\mathcal{H}_t(w,t')$ the word w must be active at time frame t'.

2.2. Naive Bayes classification model

We proposed a *smoothed naive Bayes* classification model [2] to profitably combine different predictor *scores*. We denote the class variable by c; $c = 0$ for correct and $c = 1$ for incorrect. Given a hypothesized word w and a D-dimensional vector of *scores* $\boldsymbol{x}$, the class posteriors can be calculated via the Bayes' rule as

$$P(c|\boldsymbol{x}, w) = \frac{P(c|w)\, P(\boldsymbol{x}|c, w)}{\sum_{c'} P(c'|w)\, P(\boldsymbol{x}|c', w)} \tag{6}$$

We make the naive Bayes assumption that the *scores* are mutually independent given a class-word pair [2] . Unknown probabilities are estimated by direct relative frequencies. For robustness, this *word-dependent* (specific) model is smoothed using a *word-independent* (generalized) naive Bayes model [2]. Classification is performed by classifying a word as incorrect if $P(c = 1 \mid \boldsymbol{x}, w)$ is greater that a certain threshold τ.

3. MAXIMUM ENTROPY APPROACH

In this work we will modify naive Bayes classification model proposed in [2] by estimating $P(\mathbf{x}|c, w)$ densities using a maximum entropy model. Given a class c, a hypothesized word w, and x, a generic component of the score vector $\mathbf{x}$, we can estimate $P(x|c, w)$ as a conditional maximum entropy model given by:

$$P(x/c, w) = \frac{exp\left[\sum_i \lambda_k f_i(x, c, w)\right]}{Z(c, w)} \tag{7}$$

$$Z(c, w) = \sum_{x'} exp\left[\sum_i \lambda_i f_i(x', c, w)\right] \tag{8}$$

Where $f_i(x, c, w) \in \{0, 1\}$ and λ_i are the features and the parameters of the model respectively. In order to simplify notation $f_i(x, c, w)$ will be stated from now as f_i. The rest of the model, ie. densities $P(\mathbf{x}|c, w)$ and class posteriors (6) are estimated as in [2] using naive Bayes independence assumption. Each of the features functions is related to a constrain equation in the form of conditionals expectations. Those expectations are imposed to be equal to empirical expectations, that is, the expectation of the feature with respect to an empirical distribution defined by the training corpus:

$$\sum_{x,c,w} \tilde{P}_i(x, c, w) f_i = \sum_{x,c,w} \tilde{P}(c, w) P(x/c, w) f_i \tag{9}$$

Distribution $\tilde{P}(c, w)$ is defined as:

$$\tilde{P}(c, w) = \begin{cases} \frac{N(c,w)}{N} & \text{if } N(c, w) > U(c) \\ \frac{N(c)}{N} & \text{if } N(c, w) \le U(c) \end{cases} \tag{10}$$

Where N is the size of the train set, and $U(c)$ is a threshold value which depends on the class c. We define two different approaches in order to model the distribution (7). The first model called *basic model*, is the maximum entropy version of the model described in [2] (and briefly summarized in section 2.2). We define the set of features and empirical expectations in a way that the resultant model is equivalent to that model. The second model is called *optimized model*. In this model we obtain a reduced set of features which holds the essential information provided by the basic model, with a slightly increase in performance and a dramatic reduction in the number of parameters of the model.

3.1. Basic model

In the model described in [2], distribution $P(x|c, w)$ is smoothed using an adapted version of the *absolute discounting* smoothing technique [5]. Also different smoothed distributions are used according if the count $N(c, w)$ is greater or lower than a class dependent threshold $U(c)$. It is easy to show that all those characteristics can be included in our maximum entropy model by defining the following

four sets of features f_k, f_l, f_m and f_n:

$$f_k = \begin{cases} 1 & \text{if } x = x_k,\ c = c_k,\ w = w_k, \\ & \quad N(x, c_k, w_k) > U(c_k)\ \forall\, x \\ 0 & \text{otherwise} \end{cases}$$

$$f_l = \begin{cases} 1 & \text{if } x = x_l,\ c = c_l,\ w = w_l, \\ & \quad N(x, c_l, w_l) > U(c_l) \text{ for some } x \\ 0 & \text{otherwise} \end{cases}$$

$$f_m = \begin{cases} 1 & \text{if } x = x_m,\ c = c_m,\ w = w_m, \\ & \quad N(x, c_m, w_m) \leq U(c_m) \\ & \quad N(c_m, x) > 0\ \forall\, x \\ 0 & \text{otherwise} \end{cases}$$

$$f_n = \begin{cases} 1 & \text{if } x = x_n,\ c = c_n,\ w = w_n, \\ & \quad N(x, c_n, w_n) \leq U(c_n)\ \forall\, x \\ & \quad N(c_n, x) = 0 \text{ for some } x \\ 0 & \text{otherwise} \end{cases}$$

and the empirical distribution for $N(c, w) > U(c)$:

$$\tilde{P}(x, c, w) = \begin{cases} \frac{N(x,c,w)}{N} & \text{if } N(x, c, w) > 0 \\ & \quad \forall\, x \\ \frac{N(x,c,w)-b}{N} & \text{if } N(x, c, w) > 0 \\ b\,\tilde{P}(x, c) \dfrac{\sum\limits_{x':N(x',c,w)>0} 1}{N \sum\limits_{x':N(x',c,w)=0} \tilde{P}(x', c)} & \text{if } N(x, c, w) = 0 \end{cases} \tag{11}$$

$$\tilde{P}(x, c) = \begin{cases} \dfrac{N(x, c)}{N} & \text{if } N(c, x) > 0\ \forall\, x \\ \dfrac{N(x, c) - b}{N \sum\limits_{x':N(x',c)>0} 1} & \text{if } N(x, c) > 0 \\ \dfrac{b}{N} \dfrac{\sum\limits_{x':N(x',c)>0} 1}{\sum\limits_{x':N(x',c)=0} 1} & \text{if } N(x, c) = 0 \end{cases} \tag{12}$$

Where b is a discount factor as defined in [5]. Constrain equations (9) corresponding to the sets of features f_k and f_l will be associated with the empirical distribution (11). Also, constrain equations f_m and f_n will be associated with distribution (12). The proposed basic model is slightly different from others smoothing models proposed in language model literature [5, 10] where higher and lower order distributions are defined according to the number of occurrences of a train sample. On one hand our model defines just one distribution (7), on the other hand the activation of a feature is not restricted to triplets (x_i, c_i, w_i) that occurred in the train set, but to the whole events space. (x, c, w). Thus, a triplet that never occurred in the train set, still can activate some feature of the set defined by f_m or f_n.

3.2. Optimized model

As said, the main goal of this model is to provide the minimal set of constrains that hold the most important characteristics of the model. We first hypothesize that the word w is not essential in the model to estimate the class c while, on the contrary, the score x is of significant importance. Then, in order to incorporate this hypothesis to the model we define the set of features: f_a, which activates with all the occurrences of $(c = 0, x)$ in the space of events, f_b which is similar to f_a except that this feature activates with the occurrences $(c = 1, x)$ in the train set, and finally f_c which has an additional requirement over the frequency of a word w.

$$f_a = \begin{cases} 1 & \text{if } x = x_a,\ c = 0 \\ 0 & \text{otherwise} \end{cases}$$

$$f_b = \begin{cases} 1 & \text{if } x = x_b,\ c = 1,\ N(w) > 0 \\ 0 & \text{otherwise} \end{cases}$$

$$f_c = \begin{cases} 1 & \text{if } x = x_c,\ c = 0,\ N(w) \leq F_{max} \\ 0 & \text{otherwise} \end{cases}$$

Finally, in an effort to emphasize the discrimination properties of the score to classify classes c we define a set of four features called f_1, f_2, f_3 and f_4 given by:

$$f_1 = \begin{cases} 1 & \text{if } x \leq u_{x1},\ c = 0,\ w = w_1,\ N(0, w_1) > U(0) \\ 0 & \text{otherwise} \end{cases}$$

$$f_2 = \begin{cases} 1 & \text{if } x > u_{x1},\ c = 0,\ w = w_2,\ N(0, w_2) > U(0) \\ 0 & \text{otherwise} \end{cases}$$

$$f_3 = \begin{cases} 1 & \text{if } x \leq u_{x2},\ c = 1,\ w = w_3,\ N(1, w_3) > U(1) \\ 0 & \text{otherwise} \end{cases}$$

$$f_4 = \begin{cases} 1 & \text{if } x > u_{x2},\ c = 1,\ w = w_4,\ N(1, w_4) > U(1) \\ 0 & \text{otherwise} \end{cases}$$

Where u_{x1} and u_{x2} are threshold values determined from histograms of the distribution of the score x when $c = 0$ and $c = 1$ respectively. Both sets of optimized features will be associated to the corresponding empirical distribution $\tilde{P}_i(x, c, w)$. defined by (11) and (12).

4. EXPERIMENTAL STUDY

We carried out experiments using two different corpora. One is the *Traveler task* (Eu-I), a Spanish speech corpus of person-to-person communication utterances at the reception desk of a hotel [6]. The other is the *FUB task* (Eu-II), an Italian speech corpus of *phone* calls to the front desk of a hotel [7]. Main features of the (disjoint) training and test sets, for both corpora, acquired in the context of the EUTRANS project [6, 7], are summarized in table 1.

Table 1. *Eu-I and Eu-II speech corpus.*

	Eu-I task		Eu-II task	
	training	test	training	test
# speakers	20	12	276	24
# run. words	13,728	3,390	52,511	5,381
# vocabulary	683	–	2,459	–
bigram perplex.	–	6.8	–	31

In order to evaluate classification accuracy we use the well known measure AROC which is the area under the *Receiver Operating Characteristic* (ROC) curve divided by the area of a worst case diagonal ROC curve. An AROC value of 2.0 would indicate that all words can be correctly classified. In order to compare the maximum entropy approach to alternative confidence estimation

criteria, we have used the predictor *scores* described in section 2.1.1 directly as confidence measure such as is proposed in [1]. We used *PostMax* since it performs slightly better than *PostMed* [1]. Table 2 shows the comparative AROC values using single and combined *scores*. Column labeled NB correspond to the naive bayes classifier model described in section 2.2. The others columns correspond to maximum entropy models both, basic (ME-BM) and optimized (ME-OM) models. Table 3 shows the number of parameters of the corresponding distribution for both maximum entropy models.

Table 2. *AROC values for single and combined scores.*

	Predictor Score	NB	ME-BM	ME-OM
Eu-I	PostMax	1.89	1.89	1.90
	WTS+PostMax	1.91	1.91	1.91
Eu-II	PostMax	1.75	1.75	1.76
	AS+AcSc+WTS+PostMed	1.84	1.84	1.84

Table 3. *Number of model parameters.*

	Eu-I		Eu-II	
	BM	OM	BM	OM
PostMax	12348	93	26544	401
WTS+PostMax	38892	186	–	–
AS+AcSc+WTS+MedMax	–	–	104912	1193

5. DISCUSSION

Table 2 shows the performance of different *scores*. We can see that the combination of different *scores* produces better performance than the use of a single *score*. This is most significant for Eu-II corpora. We can also see that maximum entropy basic model has exactly the same performance than the naive Bayes classifier. This is an expected result, because as said, the features of this model were designed to meet this requirement. Estimation of the parameters in maximum entropy models is usually performed using the GIS algorithm [9]. In our basic model it requires no more than five iterations for convergence, so, the CPU time involved in both model is approximately the same. It is important to note that features of the maximum entropy model were designed to give a consistent model, so no convergence problems were found. Then we conclude that no significance improvement is obtained by the use of the basic model from a practical point of view. The optimized model also produces similar results than naive Bayes classifier and basic model, although we have a small improvement for the single score model. The main difference, as can be seen at table 3 is that the number of parameters of the optimized model is nearly two order of magnitude lower than the basic model. This difference not only dramatically decrease the CPU time requirements of the model, but also permit us to gain some insight into the model structure. A detailed analysis of the performance of the optimized model, shows that performance does not significantly decrease if we only hold features f_a and f_b and discard the rest of the features. This shows that classification is performed mainly based on the score x and that is nearly independent of the word w. The rest of the features, highlight other minor details of the model that also increase the performance. They include the limits of the classification threshold of score x, (features f_1 to f_4), and the frequency of the words (feature f_c). We can conclude that the optimized model is not only important from the computational efficiency of the model, but also from the possibility of capture just essential information of the model through selected features.

6. CONCLUSIONS

We have presented two maximum entropy models for estimating confidence measures for speech recognition. The basic model is devoted to reproduce the main goals of a naive Bayes classifier which has proved good performance for this problem. We also proposed an optimized model which preserves only the essential information needed to perform classification without loose of performance. The basic model also includes a novel method to handle unseen data, defining a unique distribution for unseen and seen data. Special features, were defined in both models for handling both unseen data and the most relevant information of the model.

7. REFERENCES

[1] F. Wessel, R.Schlüter, K.Macherey and H.Ney. Confidence measures for large vocabulary continuous speech recognition. *IEEE Trans. on Speech and Audio Processing*, 9(3):288–298, 2001.

[2] A. Sanchis, A. Juan, and E. Vidal. "Improving utterance verification using a smoothed naive bayes model". ICASSP'2003, vol. 1, pp. 592–595, 2003.

[3] A. Sanchis, A. Juan and E. Vidal. Estimating confidence measures for speech recognition verification using a smoothed naive Bayes model. IbPRIA 2003 Proceedings. Lecture Notes in Computer Science LNCS 2652, pp. 910–918, 2003.

[4] A. Sanchis, A. Juan and E. Vidal. New features based on multiple word graphs for utterance verification. 8th International Conference on Spoken Language Processing, 2545-2548, 2004.

[5] H. Ney, S. Martin, and F. Wessel, "Statistical language modeling using leaving-one-out," *Young, S. and Bloothoft, G., editors, Corpus Based Methods in Language and Speech Processing*, pp. 174–207, 1997.

[6] J.C. Amengual, J.M. Benedí, F. Casacuberta, M.A. Castaño, A. Castellanos, V.M. Jiménez, D. Llorens, A. Marzal, M. Pastor, F. Prat, E. Vidal, and J.M. Vilar. The EuTrans-I speech translation system. *Machine Translation*, vol. 15, pp. 75–103, 2000.

[7] F. Casacuberta, H. Ney, F. J. Och, E. Vidal, J. M. Vilar, S. Barrachina, I. Garcia-Varea, D. Llorens, C. Martinez, S. Molau, F. Nevado, M. Pastor, D. Picó and A. Sanchis. Some approaches to statistical and finite-state speech-to-speech translation. Computer Speech and Language, vol. 18, pp. 25–47, 2004.

[8] S. C. Martin, H. Ney, and J. Zaplo, "Smoothing methods in maximum entropy language modeling," in *ICASSP 99*, vol. 1, pp. 545–548, 1999.

[9] V. D. P. Adam L. Berger, Stephen D. Pietra and J. Lafferty, "A maximum entropy approach to natural language processing," *Computational Linguistics*, vol. 22, no. 1, 1996.

[10] S. C. Martin, H. Ney, and C. Hamacher, "Maximum entropy language modeling and the smoothing problem," *IEEE Trans. on Speech and Audio Processing.*, vol. 8, pp. 626–632, September 2000.

[11] R. Rosenfeld, "A maximum entropy approach to adaptive statistical language modeling," *Computer Speech and Language*, vol. 10, pp. 187–228, 1996.