Automatic Speech Recognition

We performed experiments on the TIMIT database with the standard splitting into train, development and test data. The signal was preprocessed using the procedure described in (Sha & Saul, 2007). There are 3696 utterances and over 1 million frames in the training set. A left-right HMM with one to three states and Gaussian mixture probability densities was build for each of 48 phonetic classes. We followed standard conventions in mapping the 48 phonetic labels down to 39 broader phone categories and error rates were computed as the sum of substitution, deletion, and insertion error rates from the alignment process.

We naturally compared our algorithms with a non discriminant system (MLE) (trained with the HTK Toolkit). In addition this MLE system is used during the training of discriminant systems both for initialization and for regularization. Actually we used the regularization term $\frac{\lambda}{2}\|\mathbf{w}-\mathbf{w}^{MLE}\|^2$ which experimentally performs slightly better than $\frac{\lambda}{2}\|\mathbf{w}\|^2$. Moreover, using $\|\mathbf{w}-\mathbf{w}^{MLE}\|^2$ for regularization term leads to a bigger optimal value of λ than using $\|\mathbf{w}\|^2$, which reduces considerably the number of training steps (and also the number of cutting planes required to approximate well the objective function). We implemented two variants of our method (non-convex optimization or NCO), one uses the hard-max (NCO-H) and the other one (NCO-S) uses the soft-max version over all possible labellings (see Eq. (4),(9)), this latter version is implemented with a Forward-Backward procedure. Also, we used the hamming distance for $\Delta(\mathbf{y}^i, \mathbf{y})$.

Experimentally NCO-S is about 10 times slower than NCO-H which is 2 times slower than MLE training, we give hints now. Actually learning cost mainly decomposes into two terms, computing frames probabilities and dynamic programming. In exprimental settings such as in speech recognition the first term dominates and is similar for NCO and MLE methods if training sentences include many different phones. Besides, to reach a *gap* $< 1\%$, NCO-H requires about two times more iteration than MLE requires to converge. Finally NCO-H training is 2 times slower than MLE.

We compared our methods to three competitive discriminant methods, the large margin convex formulation of (Sha & Saul, 2007) (named Oracle), and two benchmark discriminant methods, Conditional Maximum Likelihood (CML) and Minimum Classification Error (MCE). Table 1 shows phone error rates of all these methods for one-state HMM per phone. Note that Oracle, MCE and MMI results are taken from (Sha, 2006) and correspond to the same experimental setting. These results clearly show first that discriminant approaches significantly outperform MLE training, and second that large margin approaches (NCO and Oracle) significantly outperform the two other discriminant methods. Note also that the two variants of our method NCO-H and NCO-S perform similarly. Since NCO-H is much faster we report only NCO-H results in the following. Table 2 shows results with a few states per left-right phone HMM, for the two most efficient techniques (NCO and Oracle) only. Note that (Sha, 2006) only report results for 3 states HMM with a small number of gaussians. As may be seen in these experiments the oracle method is not able to exploit the increasing complexity of the models while our method can take advantage of the number of states to reach lower error rates. We believe that this success comes from the original non-convex formulation.

Table 1. Phone error rates with single state phone HMMs ($N = 1$) and mixtures of M Gaussian laws per state.

M	MLE	NCO-H	NCO-S	Oracle	CML	MCE
1	44.75	31.44	31.02	31.2	36.4	35.6
2	39.54	29.70	30.21	30.8	34.6	34.5
4	36.06	29.13	29.30	29.8	32.8	32.4
8	34.46	28.29	29.11	28.2	31.5	30.9

Table 2. Phone error rates with multi-state phone HMMs.

N	M	MLE	NCO-H	Oracle(Sha, 2006)
2 states	1	38.21	29.57	Not Available
2 states	2	34.14	27.99	NA
2 states	4	32.00	27.67	NA
2 states	8	31.25	27.58	NA
3 states	1	36.70	28.70	37.8
3 states	2	31.92	27.93	32.6
3 states	4	30.28	27.40	NA
3 states	8	29.55	27.61	NA

On-line Handwriting Recognition

On-line handwriting signals are temporal sequences of the position of an electronic pen captured through a digital tablet. We used a part of the Unipen international database with a total of 15k digit samples, 5k samples for training and 10k samples for testing. We trained a five states left-right CDHMM for each digit.

Table 3 reports classification error rates of three systems, namely MLE, the Oracle method and NCO-H. Again, one can see that our method reaches the best results whatever M the number of Gaussian in Gaussian mixtures. NCO-H is shown to significantly outperform the Oracle based method showing that our algorithm has been able here too to efficiently learn from partially labeled training samples.