Phonetic Segment Rescoring using SVMs

Yeojin Kim
2013 Beckman Institue
University of Illinois at Urbana-Champaign
Illinois 61801

Mark Hasegawa-Johnson
2011 Beckman Institute
University of Illinois at Urbana-Champaign
Illinois 61801

Abstract

We used SVM to rescore the output of an HMM speech recognizer. We focused on confusable phone pairs to improve recognition rates and used the confusion matrix in order to choose confusable pairs. We performed experiments using parallel SVMs to determine which frames are most useful for rescoring in each context.

1 Introduction

Hidden Markov Models (HMM) are used in speech recognition widely as a model with automatic time alignment. A fundamental limit of HMM is the trade-off between complexity and accuracy of Gaussian Mixture models (GMM) (Rabiner, 1989). Although GMM has good performance in terms of expressing speech signals, an accurate GM model of the speech PDF requires several dozen mixture elements, therefore requiring hundreds of hours of speech data to train.

A Support Vector Machine (SVM) produces an optimal decision hyper-plane for binary classification (Burges, 1998). However, it has a disadvantage; it does not consider time alignment. Combining HMM with time property and SVM providing an optimal decision hyper-plane is expected to improve speech recognition error rates.

Ganapathiraju and Picone (2000) described the use of SVM within the framework of HMM based speech recognition and estimated a warping function that maps SVM distances to posterior probabilities. Their approach is to divide the segment into three regions in a set ratio and construct a composite vector from the mean vectors of the three regions. Composite vectors are generated for each of the segments and posterior probabilities are hypothesized that are used to find the best word sequence using the Viterbi decoder.

However, it remains any difficulties to discriminate confusable phone pairs. Admittedly, these confusable pairs have much do with increasing phone recognition error rates. It seems hard to reduce the error rates by only segmental features of them without additional information revealing characters of each phone. This paper suggests a way to choose most confusable phones by use of the confusion matrix and find most useful frames which include distinguishable features for these pairs. Our approach is to introduce neighbor frames out of the segments to SVM and the result was better than using only segmental features.

2 Support Vector Machines

SVM is a machine learning method to perform pattern recognition between two classes by finding a decision surface that has maximum distance from the closest points in the training set, which are termed support vectors. The decision function has the form

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i x_i \cdot x + b \qquad (1)$$

where the coefficients α_i and the b are the solutions of the quadratic programming problem and N is the number of support vectors. The vectors x_i that satisfy $y_i(w \cdot x_i + b) = 1$ as a decision function are called support vectors.

The entire construction can be extended to the case of nonlinear separating surfaces. Each point x in the input space is mapped to a point $z = \Phi(x)$ of higher dimensional space, called the feature space, where the data are separated by a hyper-plane. The key property in this construction

is that the mapping of $\Phi(\cdot)$ is subject to the condition that the dot product of two points in the feature space, $\Phi(x)\cdot\Phi(y)$, can be rewritten as a kernel function $K(x, y)$. Then the decision function has the form

$$f(x) = \sum_{i=1}^{N} \alpha_i y_i K(x_i, x) + b \tag{2}$$

An important family of kernel functions is the homogeneous polynomial kernel $K(x, y) = (x \cdot y)^d$ and the radial basis function (RBF) kernel

$$K(x, y) = \exp(-\sigma^{-2}\|x - y\|^2) \tag{3}$$

3 Parallel Support Vector Machines

We need to find some way to represent the context-dependence of each phone in the lattice. For example, consonants influence the spectral transition into the following; phonetic distinguishing information is not limited to frames covered by the phone. Therefore, we can reduce error rates by including or excluding neighbor frames according to the context.

There are several ways to select frames for an SVM. We divided a phonetic segment to several parts in order to know which frames are most useful. Our experiments show that we get lowest error rates when splitting a phonetic segment to several SVM classifiers rather than classifying segment with one SVM. Each frame goes through the related SVM and the discriminant functions from the SVMs are summed and used to determine from which phone the segment is. These parallel SVMs for a confusable pair may have different importance and the difference could be reflected by giving a proper weight to each SVM.

4 Confusion Network

HTK produces a lattice from training data and the lattice is scored to select the correct phones (Young et al., 2002). The lattice can be "pinched" and rescored using posterior probability in order to further reduce word error rate (Mangu et al., 2000).

Phone errors in the first pass recognizer output can be summarized in a confusion matrix. So we use the confusion matrix in order to choose confusable pairs. Many phone errors involve phones which are in the same phonetic group and have almost the same duration. If we can limit the number of comparable phones, it is possible to train an SVM for each pair of compared phones. In this confusion matrix would be a way to limit the population of phones rationally. If we can discriminate these pairs correctly, we can rescore the lattice, yielding higher recognition rates. Therefore it is valuable to provide a finer discriminant function as a rescoring factor with time information provided by the branch of the lattice. This paper introduces SVMs as the discriminant functions to classify confusable phones in the lattice output of a speech recognizer.

5 Experiments

We considered three cases for consonants (figure 1). The first is the case of consonant with preceding vowel (PV) and the second is the case of consonant with following vowel (FV). Finally, the third is the case of consonant without any neighbor vowel (NV). We selected RBF kernel and got the optimal parameters of SVM by experiments. Table 1-3 show results of experiments distinguishing /f/ and /s/; /f/ and /s/ were the most confused pair in the output of our LPCC-HMM NTIMIT phone recognizer. Table 2 shows that we can get better results by using frames of the vowel part, FV(3). The parallel SVMs covering PV or NV part have the lowest error rate in table 1 and table 3.

The total error rate is 18.24% when the total sample number is 7269 (f: 963, s: 6306). This is 2.34% lower than non-context error rate, 20.58%.

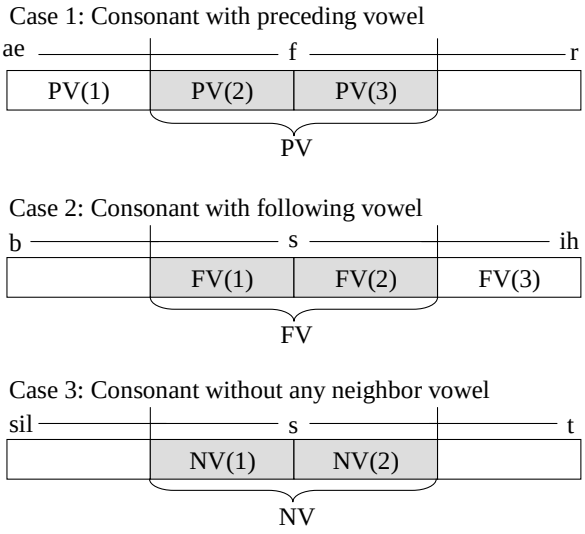

Figure 1. Splitting phonetic segment for SVM

	PV(1)	PV(2)	PV(3)	PV
/f/	0.3227	0.2173	0.3323	0.1518
/s/	0.4001	0.3236	0.1897	0.2448
Avg.	0.3891	0.3086	0.2093	0.2317
	PV(1+2)	PV(1+3)	PV(2+3)	PV(1+2+3)
/f/	0.2444	0.2460	0.1693	0.2029
/s/	0.3515	0.2367	0.2051	0.2516
Avg.	0.3363	0.2380	0.2001	0.2448

Table 1. Error rates of SVM for discrimination of /f/ and /s/ with preceding vowel (Case 1)

	FV(1)	FV(2)	FV(3)	FV
/f/	0.2368	0.3722	0.2632	0.2556
/s/	0.3096	0.1324	0.2500	0.1850
Avg.	0.2970	0.1738	0.2523	0.1972
	FV(1+2)	FV(1+3)	FV(2+3)	FV(1+2+3)
/f/	0.2256	0.1842	0.2632	0.1917
/s/	0.1959	0.2445	0.1865	0.1795
Total	0.2010	0.2341	0.1997	0.1816

Table 2. Error rates of SVM for discrimination of /f/ and /s/ with following vowel (Case 2)

	NV(1)	NV(2)	NV(1+2)	NV
/f/	0.3380	0.2113	0.2394	0.1831
/s/	0.1791	0.1455	0.1161	0.1243
Avg.	0.1878	0.1491	0.1229	0.1275

Table 3. Error rates of SVM for the discrimination of /f/ and /s/ without vowel (Case 3)

6 Conclusions

At present, we are constructing parallel SVMs for a large number of confusable pairs selected from the HMM confusion matrix. Our future work is to incorporate these parallel SVMs for lattice rescoring.

References

L. Mangu, E. Brill and A. Stolcke, “Finding consensus in speech recognition: word error minimization and other applications of confusion networks,” Computer, Speech and Language, 14(4): 373-400, 2000.

C. J. C. Burges, “A tutorial on support vector machines for pattern recognition,” Data Mining and Knowledge Discovery, 2(2): 955-974, 1998.

L. R. Rabiner, “A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition,” Proceedings of the IEEE, vol. 77, no. 2, Feb. 1989.

S. Young, G. Evermann, T. Hain, D. Kershaw, G. Moore, J. Odell, D. Ollason, D. Povey, V. Vlatchev and P. Woodland, “The HTK Book (for HTK version 3.2.1),” 2002.

A. Ganapathiraju and J. Picone, “Hybrid SVM/HMM architectures for speech recognition,” Neural Information Processing Systems, 2000.