Infinite Support Vector Machines in Speech Recognition

Jingzhou Yang, Rogier C. van Dalen and Mark Gales

Department of Engineering, University of Cambridge
Trumpington Street, Cambridge, CB2 1PZ, UK
{jy308,rcv25,mjfg}@eng.cam.ac.uk

Abstract

Generative feature spaces provide an elegant way to apply discriminative models in speech recognition, and system performance has been improved by adapting this framework. However, the classes in the feature space may be not linearly separable. Applying a linear classifier then limits performance. Instead of a single classifier, this paper applies a mixture of experts. This model trains different classifiers as experts focusing on different regions of the feature space. However, the number of experts is not known in advance. This problem can be bypassed by employing a Bayesian non-parametric model. In this paper, a specific mixture of experts based on the Dirichlet process, namely the infinite support vector machine, is studied. Experiments conducted on the noise-corrupted continuous digit task AURORA 2 show the advantages of this Bayesian non-parametric approach.

Index Terms: generative feature space, Bayesian non-parametric, Dirichlet process, mixture of experts, infinite support vector machines

1. Introduction

In previous work [1, 2, 3, 4], a variety of discriminative models based on generative feature spaces were studied and experiments showed improvement compared with well-trained hidden Markov models (HMMs) [5]. The main advantage of using generative feature spaces, which are built on generative models, is that model-based adaptation and compensation can be implemented on the generative models in the process of evaluating the feature spaces. This can enhance the noise robustness of the recognition system. This paper focuses on a certain type of Bayesian non-parametric model [6] named the infinite support vector machine (iSVM) [7] in speech recognition based on generative feature spaces derived from vector Taylor series (VTS) [8] compensated HMMs.

Speech recognition can be considered as a problem of classifying sequential audio data (e.g. vectors of MFCCs). Normally, the sequential data vary in length, but classifiers such as SVMs can only handle data with fixed dimension. In order to bridge this gap, a generative model (e.g. an HMM) can be applied on the sequential data to derive features with fixed dimension [9]. In previous work [2, 3], one linear SVM classifier was adopted on the feature space may be not linearly separable. Rather than applying a single linear classifier on the feature space, it is more reasonable to utilise a mixture of experts [10] which makes an ensemble decision by all experts with various weights determined by the region of space. Figure 1 illustrates an single classifier and a mixture of experts on the feature space.

Although the kernel trick could be applied on the SVM to yield a non-linear decision boundary, it might be problematical to choose the type of kernel, and the number of support vectors might be large which leads to inefficiency in testing. Thus, the kernel trick is not considered in this paper, even if the kernel trick could be applied on the mixture of experts as well.

In terms of the mixture of experts, it is hard to set the number of experts to fit with the training data under a parametric framework. In contrast, the model complexity in a Bayesian non-parametric model is treated as one set of model parameters, and the posterior distribution of the complexity can be inferred. Then, the model complexity can be integrated out when making predictions, namely the model is averaged over all possible complexities. Thus, a Bayesian non-parametric model becomes a better choice in tackling the problem of selecting model complexity. In [11], a Bayesian non-parametric model named sticky hierarchical Dirichlet process hidden Markov model (HDP-HMM) [12] was introduced to infer the unknown number of speakers in a speech diarisation task, and state-of-art performance was achieved. In this paper, one example of Dirichlet process (DP) mixture of experts named iSVM is implemented to resolve the problem of the unknown number of experts.

This paper is organised as follows. Section 2 discuss various features derived from generative models. The mixture-of-experts model is discussed in section 3, and its non-parametric counterpart called DP mixture of experts is introduced in section 4. Section 5 details a specific example of the DP mixture of experts model, the iSVM. Finally, the experimental results and conclusions are presented in section 6 and 7.

2. Features

In speech recognition, the speech utterances normally vary in length. In order to handle the length variation of the observations, generative models can be utilised to map the sequential data (on the input space) with various length to feature vectors

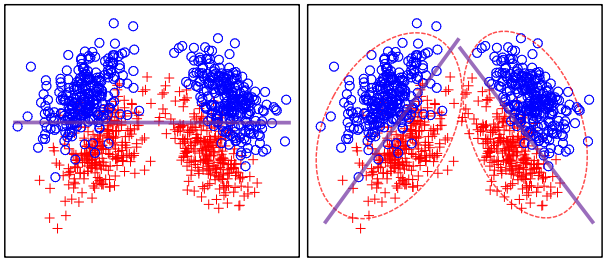

Figure 1: *The single classifier and mixture of experts on the feature space.*

This work was partially supported by EPSRC Project EP/I006583/1 within the Global Uncertainties Programme and DARPA under the RATS program. The paper does not necessarily reflect the position or the policy of US Government and no official endorsement should be inferred. J.Yang would like to thank Austin Zhang and Eric Wang for the help in research and daily life.

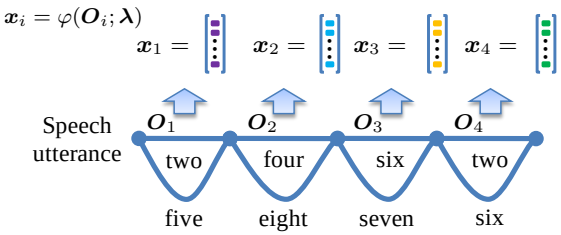

Figure 2: *The process of generating the feature vectors.*

(on the feature space) with fixed dimension. Assume the speech observations are $\mathcal{O} = \{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_I\}$, where $\{\boldsymbol{O}_1, \ldots, \boldsymbol{O}_I\}$ are one possible segmented data of the observations $\mathcal{O}$ (e.g. the 1-best hypothesis of the lattice), and each segment $\boldsymbol{O}_i$ specifies a word/phone/sub-phone. One possible utterance is illustrated in Figure 2. Given a segment $\boldsymbol{O}_i$, the log-likelihood feature space can be described as follows [4]:

$$\varphi^l(\boldsymbol{O}_i; \boldsymbol{\lambda}) = \frac{1}{T_i} \begin{bmatrix} \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_1})\right) \\ \vdots \\ \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_\text{L}})\right) \end{bmatrix}_{\text{L}\times 1} \tag{1}$$

where $\{\tilde{w}_1, \ldots, \tilde{w}_\text{L}\}$ are all the unique classes, or vocabularies, T_i is the number of frames in the observation $\boldsymbol{O}_i$, and $p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_l})$ is the likelihood of the generative model parameters corresponding to class $\tilde{w}_l$ given $\boldsymbol{O}_i$. Figure 2 illustrates the process of deriving features from an utterance. For simplicity, $\varphi(\boldsymbol{O}_i; \boldsymbol{\lambda})$ is written as $\boldsymbol{x}_i$.

A more general form of generative feature space is the derivative feature space, which not only includes the log-likelihood of the parameters but also incorporates the derivatives of the log-likelihood with respect to the parameters of the generative models. If only the first order of the derivatives is considered, the feature space can be described as follows:

$$\varphi^d(\boldsymbol{O}_i; \boldsymbol{\lambda}) = \frac{1}{T_i} \begin{bmatrix} \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_1})\right) \\ \nabla_{\boldsymbol{\lambda}_{\tilde{w}_1}} \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_1})\right) \\ \vdots \\ \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_\text{L}})\right) \\ \nabla_{\boldsymbol{\lambda}_{\tilde{w}_\text{L}}} \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_\text{L}})\right) \end{bmatrix} \tag{2}$$

According to previous work [3, 4], the derivatives with respect to the mean have better discrimination than the derivatives with respect to other parameters. The derivative with respect to the mean of the component m can be described as follows:

$$\nabla_{\boldsymbol{\mu}_m} \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_1})\right) = \sum_t \gamma_m(t) \boldsymbol{\Sigma}_m^{-1}(\boldsymbol{o}_t - \boldsymbol{\mu}_m) \tag{3}$$

where $\gamma_m(t)$ is the posterior probability of component m generating $\boldsymbol{o}_t$.

In order to adapt the generated features to the target noise condition, state-of-art model-based adaptation and compensation technology can be implemented on generative models in the process of deriving the features. VTS compensation [8] is adopted in our work. The parameters of convolutional noise and additive noise are estimated on each utterance to maximise the likelihood of the HMM system.

3. Mixture of experts

Rather than making prediction on the whole feature space from a single model, another method is choosing different models to make prediction according to different features, and the choice

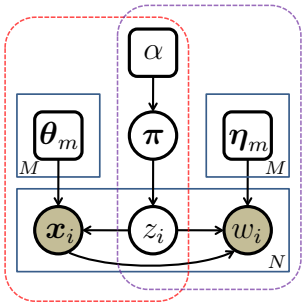

Figure 3: *The graphical model of the mixture of experts.*

of the model is input-dependent. When the choice of the model is a hard decision, this framework is known as a *decision tree*. When the choice of each model is given a probability depending on the input, this is known as a *mixture of experts* [13]:

$$P(w|\boldsymbol{x}, \boldsymbol{\theta}_a, \boldsymbol{\eta}_a) = \sum_{m=1}^{M} P(z = m|\boldsymbol{x}, \boldsymbol{\theta}_a) P(w|\boldsymbol{x}, \boldsymbol{\eta}_m) \tag{4}$$

where $\boldsymbol{x}$ is the variable in the feature space $\varphi(\boldsymbol{O}_i, \boldsymbol{\lambda})$ given in section 2, w is the target variable, M is the number of experts. $P(z = m|\boldsymbol{x}, \boldsymbol{\theta}_a)$ is the *gating network* which assigns probabilities to different experts according to the input $\boldsymbol{x}$, and $P(w|\boldsymbol{x}, \boldsymbol{\eta}_m)$ is the m^{th} *expert*, which is a discriminative model. $\boldsymbol{\theta}_a = \{\boldsymbol{\theta}_1, \ldots, \boldsymbol{\theta}_M\}$ are the parameters of the gating network, and $\boldsymbol{\eta}_a = \{\boldsymbol{\eta}_1, \ldots, \boldsymbol{\eta}_M\}$ are the parameters of the M experts. If the gating network is given from the component posteriors of the mixture model and the number of experts is given M, the graphical model of the mixture of experts is illustrated in Figure 3, and the corresponding generative process of this model can be described as follows:

$$\boldsymbol{\pi} \sim \text{Dirichlet}(\alpha) \qquad z_i \sim \text{Categorical}(\boldsymbol{\pi}) \tag{5}$$

$$\boldsymbol{x}_i \sim p(\boldsymbol{x}|\boldsymbol{\theta}_{z_i}) \qquad w_i \sim P(w|\boldsymbol{x}_i, \boldsymbol{\eta}_{z_i}) \tag{6}$$

where $\boldsymbol{\pi} = \{\pi_1, \ldots, \pi_M\}$ is a discrete distribution which is drawn from a *symmetric Dirichlet distribution* with concentration parameter α, and z_i is the indicator variable that denotes the i^{th} datum is associated with which expert. Categorical($\boldsymbol{\pi}$) is the *categorical distribution* which is the generalisation of the *Bernoulli distribution* with M possible outcomes. $\boldsymbol{\theta}_{z_i}$ are the parameters of the component indicated by z_i, and $\boldsymbol{\eta}_{z_i}$ are the parameters of the expert indicated by z_i.

4. Dirichlet process mixture of experts

As a parametric model, the number of experts M needs to be fixed in advance for the mixture of experts. In order to bypass this problem of choosing model complexity, a Bayesian non-parametric version of the mixture-of-experts model, namely DP mixture of experts, is used here. In [14], the non-parametric model called infinite Gaussian mixture model (iGMM) is derived by setting the number of components to infinity in the Gaussian mixture model (GMM). Similarly, in this paper, the corresponding non-parametric counterpart of the mixture-of-experts model is derived, when the number of experts goes to infinity $M \to \infty$ in the mixture-of-experts model.

Since $\boldsymbol{\pi}$ is a discrete probability and $z_i \sim \text{Categorical}(\boldsymbol{\pi})$, then $P(z_1, \ldots, z_i)$ has a multinomial distribution. By marginalising out $\boldsymbol{\pi}$, the following result can be derived [14]:

$$P(z_1, \ldots, z_i|\alpha) = \frac{\Gamma(\alpha)}{\Gamma(i + \alpha)} \prod_{m=1}^{M} \frac{\Gamma(N_m + \alpha/M)}{\Gamma(\alpha/M)} \tag{7}$$

where N_m is the number of data points associated with the m^{th} expert. The conditional probability of the indicator z_{i+1} given the other i indicators can be described as follows:

$$P(z_{i+1}=m|z_1,\ldots,z_i,\alpha) = P(z_{i+1}|z_1,\ldots,z_i,\alpha)|_{z_{i+1}=m} = \frac{P(z_1,\ldots,z_{i+1}|\alpha)}{P(z_1,\ldots,z_i|\alpha)}\bigg|_{z_{i+1}=m} = \frac{N_m+\alpha/M}{i+\alpha} \quad (8)$$

When the number of experts goes to infinity $M \to \infty$, the experts' weights $\boldsymbol{\pi}$ are given by the *Dirichlet process* (DP) [15], and the conditional probability of the indicator variable z_{i+1} given all the previous i indicators can be described as follows:

$$P(z_{i+1}=m|z_1,\ldots,z_i,\alpha) = \begin{cases} \frac{N_m}{i+\alpha}, \text{ where } m \text{ is an existing expert} \\ \frac{\alpha}{i+\alpha}, \text{ where } m \text{ is a new expert} \end{cases} \quad (9)$$

According to equation (9), the probability of the indicator indicating an existing component is proportional to the number of data associated with that component, and the probability of assigning to a new component is proportional to α. The process of assigning the data to the components according to equation (9) is also known as the *Chinese Restaurant Process* (CRP) [16]. The CRP provides a mechanism to draw from the distribution given by a DP without specifying that distribution.

When $M \to \infty$, the DP mixture-of-experts model could be derived, then the corresponding generative process of this model can be described as follows:

$$\boldsymbol{z} \sim \text{CRP}(\alpha) \qquad \boldsymbol{\theta}_m \sim G_1, \boldsymbol{\eta}_m \sim G_2, \forall m \in \boldsymbol{z} \quad (10)$$

$$\boldsymbol{x}_i \sim p(\boldsymbol{x}|\boldsymbol{\theta}_{z_i}) \qquad w_i \sim P(w|\boldsymbol{x}_i, \boldsymbol{\eta}_{z_i}) \quad (11)$$

where $\boldsymbol{z} = \{z_1,\ldots,z_N\}$ are all the indicators, which are given by the Chinese restaurant process CRP(α) with parameter α. The parameters of the gating network $\boldsymbol{\theta}_m$ and the parameters of the experts $\boldsymbol{\eta}_m$ are given by the base distributions G_1 and G_2 respectively. The corresponding graphical model is illustrated in Figure 4.

This non-parametric model used for classification can be approximated by the samples from the posterior distribution of the model parameters:

$$P(w|\boldsymbol{x},\mathcal{D}) = \int P(w|\boldsymbol{x},\Theta)p(\Theta|\mathcal{D})d\Theta \approx \frac{1}{K}\sum_{k=1}^{K} P(w|\boldsymbol{x},\Theta^{(k)})$$

$$= \frac{1}{K}\sum_{k=1}^{K}\sum_{m=1}^{M_k} P(z=m|\boldsymbol{x},\boldsymbol{\theta}_a^{(k)})P(w|\boldsymbol{x},\boldsymbol{\eta}_a^{(k)},z=m) \quad (12)$$

where $\Theta = \{\boldsymbol{\theta}_a, \boldsymbol{\eta}_a\}$ are all the parameters of the DP mixture of experts. For this non-parametric model, the posterior distribution of the parameters $p(\Theta|\mathcal{D})$ is extremely complicated. This makes the integral in equation (12) intractable. Thus, a *Markov chain Monte Carlo* (MCMC) [17] method is used to approximate this integral. $\{\Theta^{(1)},\ldots,\Theta^{(K)}\}$ are sampled from the posterior distribution of the parameters $p(\Theta|\mathcal{D})$. Since this distribution is complicated, it is impractical to draw samples from this full joint distribution directly. Therefore, the *Gibbs sampling* [18] is implemented here to draw samples from the conditional posterior distribution of each parameter given all others $p(\boldsymbol{\theta}_m|\Theta_{-\boldsymbol{\theta}_m},\mathcal{D})$, rather than sample from the posterior distribution of the whole parameter set $p(\Theta|\mathcal{D})$, where $\Theta_{-\boldsymbol{\theta}_m}$ denotes all the parameters Θ except $\boldsymbol{\theta}_m$.

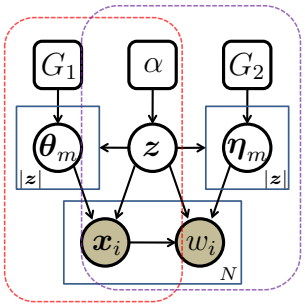

Figure 4: *The graphical model of the DP mixture of experts.*

The indicator z_i is sampled according to the follows:

$$P(z_i|\boldsymbol{x}_i, w_i, \boldsymbol{z}_{-i}, \Theta) \propto P(z_i|\boldsymbol{z}_{-i},\alpha)p(\boldsymbol{x}_i|\boldsymbol{\theta}_{z_i})P(w_i|\boldsymbol{x}_i,\boldsymbol{\eta}_{z_i}) \quad (13)$$

The first term $P(z_i|\boldsymbol{z}_{-i},\alpha)$ is given from the CRP described in equation (9). The second term is the component likelihood, and the last term is the expert's conditional likelihood. When z_i indicates an existing component, it is easy to calculate the last two terms. When z_i denotes a new component, the term $p(\boldsymbol{x}_i|\boldsymbol{\theta}_{z_i})$ can be obtained through the method discussed in [14]. The term $P(w_i|\boldsymbol{x}_i,\boldsymbol{\eta}_{z_i}) = \int P(w_i|\boldsymbol{x}_i,\boldsymbol{\eta})G_2(\boldsymbol{\eta})d\boldsymbol{\eta}$, and Monte Carlo sampling can be applied to estimate the probability as well. The parameters $\boldsymbol{\eta}_a$ of the log-linear models are obtained from the large margin criterion which is detailed in the next section. The sampling process of all other parameters is similar to the methods discussed in [14, 19, 20].

5. Infinite support vector machine

The *infinite SVM* (iSVM) was introduced in [7], where the iSVM is based on the stick-breaking representation [21] of the DP. In this paper, the iSVM based on the DP from a CRP perspective is introduced. In the previous section, the DP mixture of experts model is discussed. When each expert is defined as a multi-class SVM [22] in the DP mixture-of-experts model and the gating network is given from the component posterior of the DP mixture model, the iSVM can be derived. Since the mixture of experts is a probabilistic model, in order to derive the iSVM, each expert needs to be interpreted in a probabilistic way.

According to [2], the multi-class SVM can be interpreted probabilistically as a log-linear model with large-margin training criterion. The log-linear model can be described as follows:

$$P(w|\boldsymbol{x},\boldsymbol{\eta}_m) = \frac{\exp\left(\boldsymbol{\eta}_m^{\mathrm{T}}\Phi(\boldsymbol{x},w)\right)}{\sum_w \exp\left(\boldsymbol{\eta}_m^{\mathrm{T}}\Phi(\boldsymbol{x},w)\right)} \quad (14)$$

where $\Phi(\boldsymbol{x},w) = \boldsymbol{\delta}(w)\otimes\boldsymbol{x} = [\mathbf{0}^{\mathrm{T}},\ldots,\boldsymbol{x}^{\mathrm{T}},\ldots,\mathbf{0}^{\mathrm{T}}]^{\mathrm{T}}$ is the joint feature space, $\boldsymbol{\delta}(w) = [\delta(w-\tilde{w}_1),\ldots,\delta(w-\tilde{w}_I)]^{\mathrm{T}}$ denotes the position of the label w in the vocabulary, and $\otimes$ is the tensor product. The large-margin training criterion of the log-linear model is defined as follows [2]:

$$-\log p(\boldsymbol{\eta}_m) + \sum_{i=1}^{N_m}\left[\max_{w\neq w_i}\left\{\mathcal{L}(w,w_i) - \log\left(\frac{P(w_i|\boldsymbol{x}_i,\boldsymbol{\eta}_m)}{P(w|\boldsymbol{x}_i,\boldsymbol{\eta}_m)}\right)\right\}\right]_+ \quad (15)$$

where $\mathcal{L}(w,w_i)$ is the loss function which measures the distance between the reference w_i and label w, and $[\,\cdot\,]_+$ is the hinge-loss function. Assume the prior of $\boldsymbol{\eta}_m$ is given a Gaussian distribution $p(\boldsymbol{\eta}_m) = G_2 = \mathcal{N}(\boldsymbol{\mu}_\eta, \boldsymbol{\Sigma}_\eta)$ with mean $\boldsymbol{\mu}_\eta$

and scaled identity covariance matrix $\boldsymbol{\Sigma}_\eta = C\boldsymbol{I}$. By substituting this Gaussian prior and equation (14) into equation (15), the large margin criterion can be described as minimizing the follows:

$$\frac{1}{2C}||\boldsymbol{\eta}_m - \boldsymbol{\mu}_\eta||^2 + \sum_{i=1}^{N_m}\left[\max_{w\neq w_i}\left\{\boldsymbol{\eta}_m^{\mathrm{T}}\Phi(\boldsymbol{x}_i, w) + \mathcal{L}(w, w_i)\right\} - \boldsymbol{\eta}_m^{\mathrm{T}}\Phi(\boldsymbol{x}_i, w_i)\right]_+ \quad (16)$$

In equation (16), when the mean $\boldsymbol{\mu}_\eta = 0$, this becomes the training criterion of the multi-class SVM. In the iSVM, if there are very few data associated with a expert (N_m is small), the trained expert might lack generalisation. Thus, each expert deploys a non-zero mean $\boldsymbol{\mu}_\eta$ which is obtained from the multi-class SVM trained on the whole training set. By introducing the non-zero mean, the iSVM should retrieve multi-class SVM performance, if C is small enough. Better performance could be achieved by gradually increasing C.

Equation (16) is also known as the training criterion of the structural SVM [23, 24]. The main difference between the structural SVM and the multi-class SVM is the form of the joint feature space $\Phi(\boldsymbol{x}_i, w_i)$ which is defined by the observation-label pair. In the structural SVM, the observation-label pair is composed of the whole sentence $\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}$ and the corresponding labels $\{w_1, \ldots, w_n\}$, then the corresponding joint feature space is $\Phi\big(\{\boldsymbol{x}_1, \ldots, \boldsymbol{x}_n\}, \{w_1, \ldots, w_n\}\big)$. In contrast, the segments are treated independently in the multi-class SVM, and the pair consists of one segment $\boldsymbol{x}_i$ and the corresponding label w_i. The definition of the joint feature space is detailed in [24, 25]. Moreover, in the multi-class SVM, the loss function $\mathcal{L}(w, w_i)$ is defined as the *Kronecker delta* $\delta(w, w_i)$. In the structural SVM, the loss is set to a more refined function, say the Levenshtein distance, rather than the $0/1$ loss.

6. Experiments

In this paper, the experiments are conducted on the Aurora 2 database [26], which was designed to evaluate the performance of speech recognition algorithm in various noisy conditions. The utterances in this database are the continuous digit strings with vocabulary size 12 (one to nine, plus zero, oh and silence), and 8 real-world noise conditions have been added to the speech artificially over a variant of signal to noise ratio (SNR). The generative models (HMMs) are trained on the clean data with 8840 utterances recorded from 55 male and 55 female adults. The feature vectors used by the front-end HMMs consisted of 12 MFCCs appended with zeroth cepstrum, delta and delta-delta coefficients. The noise model for VTS compensation is estimated on each utterance. The performance of the VTS compensated HMM is listed in Table 1. The SVM and iSVM are trained on a subset of the multi-style training data containing 4 noise conditions (N2, N3 and N4) and 3 SNRs (20dB, 15dB and 10dB). All three test database, A, B and C, are used in the evaluation. In set A and B, each contains 4 noise conditions at 5 different SNRs (0dB to 20dB). There are 2 noise conditions at 5 different SNRs in set C.

In the experiments, log-likelihood feature vectors and derivative feature vectors are used, and these feature vectors are derived from the VTS compensated HMMs. To keep training with derivative feature vectors feasible, only the first element of the derivative with respect to each mean is used in this paper. All the experts (SVMs) of the iSVM share the same C, and the parameter C is tuned on the test set A. Figure 5 illustrates the WER of the iSVM on different feature spaces with various C. Since the parameter of each expert is given a Gaussian prior with mean $\boldsymbol{\mu}_\eta$ which is obtained from the multi-class SVM, the iSVM only achieves the baseline performance of the multi-class SVM when the C is small. By introducing the non-zero mean $\boldsymbol{\mu}_\eta$, the iSVM can at least achieve the performance of the multi-class SVM, and the optimised configuration can be obtained by gradually increasing C. Without the mean $\boldsymbol{\mu}_\eta$, the iSVM could have poor performance, because not all the experts associate with enough data.

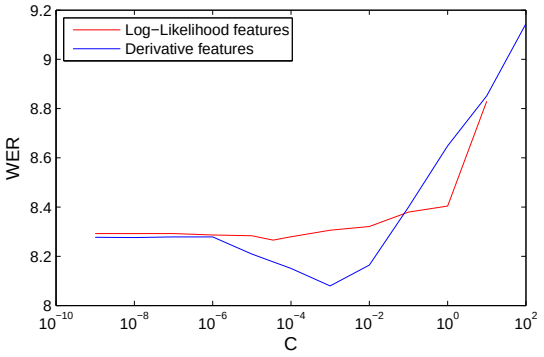

Figure 5: The performance of the iSVM on test set A with different C

System	Features	Dim	test set WER(%)			Avg
			testa	testb	testc	
HMM	MFCC	39	9.84	9.11	9.53	9.48
SVM	Log-Like	12	8.29	7.90	8.61	8.20
iSVM			8.25	7.87	8.53	8.15
SVM	Derivative	558	8.28	7.85	8.63	8.18
iSVM			8.05	7.81	8.44	8.04

Table 1: The results on Aurora 2 database

The classification criterion of the iSVM is given in equation (12), and the number of samples K is 10 here. The experimental results are listed in Table 1. All the discriminative models outperform the VTS compensated HMM baseline system. On the log-likelihood feature space and derivative feature space, the iSVM achieves better performance than the multi-class SVM. This gain is obtained by the fact that the iSVM explores the distribution of the training data and infers the number of experts, then applies different experts focus on different regions of the feature space to make an ensemble decision, rather than applying a single classifier on the whole feature space.

7. Conclusions

In this paper, a DP mixture-of-experts model based on the Chinese restaurant process has been presented, and a specific example of this type of model, the iSVM, is studied. The iSVM not only infers the number of expert in the mixture-of-experts model, but also inherits the advantages of the mixture of experts that difference experts focus on different regions of the feature space in oder to make better predictions. The experiments show the advantages of the iSVM comparing with the multi-class SVM. In this paper, the word segmentations are obtained from the 1 best hypothesis of the word lattice. Future work will study the way to optimise both the segmentations and $\boldsymbol{\eta}$, and generalise the iSVM to large vocabulary ASR by incorporating the structural SVM within the iSVM.

8. References

[1] Nathan Smith and Mark Gales, "Speech recognition using SVMs," *Advances in neural information processing systems (NIPS)*, vol. 14, pp. 1197–1204, 2002.

[2] Shi-Xiong Zhang, Anton Ragni, and Mark Gales, "Structured log linear models for noise robust speech recognition," *IEEE Signal Processing Letters*, vol. 17, pp. 945–948, 2010.

[3] Mark Gales and Federico Flego, "Discriminative classifiers with adaptive kernels for noise robust speech recognition," *Computer Speech and Language*, vol. 24, no. 4, pp. 648–662, 2010.

[4] Anton Ragni and Mark Gales, "Derivative kernels for noise robust ASR," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2011, pp. 119–124.

[5] Mark Gales and Steve Young, "The application of hidden Markov models in speech recognition," *Foundations and Trends in Signal Processing*, vol. 1, no. 3, pp. 195–304, 2008.

[6] Peter Orbanz and Yee Whye Teh, "Bayesian nonparametric models," in *Encyclopedia of Machine Learning*. Springer, 2010.

[7] Jun Zhu, Ning Chen, and Eric Xing, "Infinite SVM: a Dirichlet process mixture of large-margin kernel machines," in *Proceedings of the 28th International Conference on Machine Learning (ICML-11)*, New York, NY, USA, June 2011, pp. 617–624, ACM.

[8] Alex Acero, Li Deng, Trausti Kristjansson, and Jerry Zhang, "HMM adaptation using vector Taylor series for noisy speech recognition," in *Proceedings of ICSLP 2000*, Beijing, 2000, pp. 869–872.

[9] Mark Gales, Shinji Watanabe, and Eric Fosler-Lussier, "Structured discriminative models for speech recognition: An overview," *IEEE Signal Processing Magazine*, vol. 29, pp. 70–81, 2012.

[10] Robert A. Jacobs, Michael I. Jordan, Steven J. Nowlan, and Geoffrey E. Hinton, "Adaptive mixtures of local experts," *Neural Computation*, vol. 3, no. 1, pp. 79–87, 1991.

[11] E.B. Fox, E.B. Sudderth, M.I. Jordan, and A.S. Willsky, "A Sticky HDP-HMM with Application to Speaker Diarization," *Annals of Applied Statistics*, vol. 5, no. 2A, pp. 1020–1056, 2011.

[12] Yee Whye Teh, Michael I Jordan, Matthew J Beal, and David M Blei, "Hierarchical dirichlet processes," *Journal of the American Statistical Association*, vol. 101, no. 476, pp. 1566–1581, 2006.

[13] Christopher M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*, Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[14] Carl Edward Rasmussen, "The infinite Gaussian mixture model," in *Advances in Neural Information Processing Systems 12*. 2000, pp. 554–560, MIT Press.

[15] Yee Whye Teh, "Dirichlet processes," in *Encyclopedia of Machine Learning*. Springer, 2010.

[16] Erik B. Sudderth, *Graphical Models for Visual Object Recognition and Tracking*, Ph.D. thesis, Massachusetts Institute of Technology, Cambridge, MA, USA, 2006.

[17] Christophe Andrieu, Nando de Freitas, Arnaud Doucet, and Michael I. Jordan, "An introduction to MCMC for machine learning," *Machine Learning*, vol. 50, no. 1-2, pp. 5–43, 2003.

[18] David Wingate, "Markov chain Monte Carlo and Gibbs sampling," 2004.

[19] Radford M. Neal, "Markov chain sampling methods for Dirichlet process mixture models," *Journal of computational and graphical statistics*, vol. 9, no. 2, pp. 249–265, 2000.

[20] Carl Edward Rasmussen and Zoubin Ghahramani, "Infinite mixtures of Gaussian process experts," in *NIPS*, 2001, pp. 881–888.

[21] Jayaram Sethuraman, "A constructive definition of Dirichlet priors," *Statistica Sinica*, vol. 4, pp. 639–650, 1994.

[22] Koby Crammer and Yoram Singer, "On the algorithmic implementation of multiclass kernel-based vector machines," *Journal of Machine Learning Research*, vol. 2, pp. 265–292, 2001.

[23] Ioannis Tsochantaridis, Thomas Hofmann, Thorsten Joachims, and Yasemin Altun, "Support vector machine learning for interdependent and structured output spaces," in *Proceedings of the twenty-first international conference on Machine learning (ICML)*. ACM, 2004.

[24] Shi-Xiong Zhang and Mark Gales, "Structured SVMs for automatic speech recognition," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 21, pp. 544–555, 2013.

[25] Shi-Xiong Zhang and Mark Gales, "Structured log linear models for noise robust speech recognition," Tech. Rep. CUED/F-INFENG/TR.658, Cambridge University Engineering Department, 2010.

[26] David Pearce and Hans-Günter Hirsch, "The AURORA experimental framework for the performance evaluation of speech recognition systems under noisy conditions," in *Proceedings of the 6th International Conference on Spoken Language Processing*, 2000, pp. 29–32.