INFINITE STRUCTURED SUPPORT VECTOR MACHINES FOR SPEECH RECOGNITION

Jingzhou Yang, Rogier C. van Dalen, Shi-Xiong Zhang and Mark Gales

Department of Engineering, University of Cambridge, Cambridge, UK
{jy308,rcv25,sxz20,mjfg}@eng.cam.ac.uk

ABSTRACT

Discriminative models, like support vector machines (SVM), have been successfully applied to speech recognition, and performance has been improved. By introducing the Bayesian non-parametric version of the SVM, namely infinite SVM (iSVM), further improvement can be achieved. Since the iSVM does not consider the structure of the label, this model only can be applied to the small vocabulary recognition task. However, speech recognition is a structured classification task, where the labels are sentences and different labels may share the same structures, for example words. In order to adopt the iSVM to medium to large continuous speech recognition task, this paper studies the incorporation of the structures into the model, which is called infinite structured SVM (iSSVM), and the experiments are conducted on the noise-corrupted continuous digit task: AURORA 2.

Index Terms— Bayesian non-parametric, Dirichlet process, mixture of experts, infinite structured SVM

1. INTRODUCTION

By introducing the generative feature space [1], discriminative models, like SVM [2], have been successfully used in speech recognition. Since the features are obtained from the generative models, like hidden Markov models (HMM) [3], one main advantage of this framework is that the state-of-art speaker adaptation and noise robustness techniques [4] can be used in generating the features.

Rather than using a single classifier, the mixture-of-experts model [5, 6] deploys multiple classifiers in classification, but normally the number of experts is unknown. In order to sidestep the problem of setting experts' number, the Bayesian non-parametric framework can be used. In previous work [7], a model called infinite SVM (iSVM) has been applied to small vocabulary continuous speech recognition (CSR): Segmenting the continuous speech into segments, then each segment is classified by the iSVM similar to acoustic code breaking [8]. By using the iSVM, better performance is achieved compared with using the SVM, given that the iSVM applies all the experts in classification with different weights, which depends on the location of the data in the feature space, to make an ensemble decision.

The SVM and iSVM are un-structured models, and they only can be implemented in the small vocabulary CSR. In the medium to large vocabulary CSR task, each label corresponds to a sentence, and the possible number of labels is unlimited, but different labels may share the common structure, like words or phones. However, the un-structured models do not consider the structures of the labels, so it is impractical to use the un-structured models to model the whole utterance, for example, the possible number of classes for a 6-digit length utterance is 10^6. The structured SVM (SSVM) [9] was introduced to classify the data with structured labels. In work [10], the SSVM has been successfully used in medium to large vocabulary CSR tasks. In order to apply the mixture-of-experts framework to large vocabulary CSR, the structures must be incorporated into the model.

In this paper, a type of structured Bayesian non-parametric model called infinite structured SVM (iSSVM) is studied. Rather than using a kernel trick in the SSVM [11], which might be a problem to define the type of the kernel, the iSSVM deploys multiple SSVMs to yield a non-linear boundary. Moreover, the feature vectors used by iSSVM are given from a kernel function, this means the kernel trick can be incorporated into the kernel function. If it is necessary, the the kernel trick also can be used by each expert in the iSSVM, but the kernel trick is not considered in this paper.

This paper is organised as follows. The mixture of experts and its infinite version are discussed in Section 2. The SSVM are detailed in Section 3, and the iSSVM are introduced in Section 4. The classification and corresponding issues are discussed in Section 5. Finally, the experiments results and conclusions are given in Section 6.

2. MIXTURE OF EXPERTS

The mixture of experts applies multiple models focusing on different regions of the feature space, and the probabilities of choosing the models are input dependent. The framework of the mixture of experts with M experts is illustrated in Fig 1.

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 Eric Wang and Jeff Chen for the help in research.

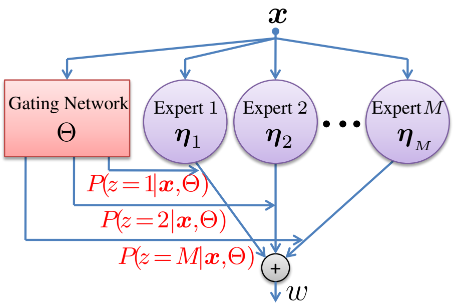

Fig. 1. *The framework of mixture-of-experts model*

The model also can be described as follows:

$$P(w|\boldsymbol{x},\Theta,H) = \sum_{z\in\mathbb{Z}} P(z|\boldsymbol{x},\Theta)P(w|\boldsymbol{x},\boldsymbol{\eta}_m) \qquad (1)$$

where $\mathbb{Z}$ is the indicator set: $\mathbb{Z} = \{1,\dots,M\}$, and the term $P(z = m|\boldsymbol{x},\Theta)$ is called *gating network*, which gives the probabilities of choosing different experts according to the input $\boldsymbol{x}$. z is the indicator variable which denotes the input $\boldsymbol{x}$ associated with which expert, and Θ is the parameter set of the gating network. The second term $P(w|\boldsymbol{x},\boldsymbol{\eta}_m)$ is the m^{th} *expert* with parameter $\boldsymbol{\eta}_m$. H is the parameter set for all the experts, and w is the class label.

When the number of experts in the mixture of expert goes to infinite, namely $M \to \infty$, and the gating network is given from a Dirichlet process (DP) mixture model [12, 13], the DP mixture of experts [7] can be derived, and it has the same form as equation (1), but the indicator set $\mathbb{Z}$ consists of infinite values: $\mathbb{Z} = \{1,2,\dots,\infty\}$.

3. STRUCTURED SVM

According to [10], the SSVM can be considered as a log-linear model with large-margin training. The log-linear model can be described as follows:

$$P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\boldsymbol{\eta}) = \frac{\exp\left(\boldsymbol{\eta}^{\mathsf{T}}\Phi(\mathcal{O},\mathcal{W};\boldsymbol{\lambda},\rho)\right)}{\sum_{\mathcal{W}',\rho'}\exp\left(\boldsymbol{\eta}^{\mathsf{T}}\Phi(\mathcal{O},\mathcal{W}';\boldsymbol{\lambda},\rho')\right)} \qquad (2)$$

where $\boldsymbol{\eta}$ is the model parameter of log-linear model. $\mathcal{W}$ is the word sequence corresponding to the utterance $\mathcal{O}$ given the segmentation ρ. Giving the segmentation ρ, the utterance and word sequence can be further described as $\mathcal{W} = \{w_1,\dots,w_{I_\rho}\}$ and $\mathcal{O} = \{\boldsymbol{O}_1,\dots,\boldsymbol{O}_{I_\rho}\}$, where w_i is a word, $\boldsymbol{O}_i$ is a segment and I_ρ is the number of segments. $\Phi(\mathcal{O},\mathcal{W};\boldsymbol{\lambda},\rho)$ is the *joint feature space*, which can be described as follows [14]:

$$\Phi(\mathcal{O},\mathcal{W};\boldsymbol{\lambda},\rho) = \frac{1}{T}\begin{bmatrix} \sum_{i=1}^{I_\rho}\delta(w_i,\tilde{w}_1)\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda}) \\ \vdots \\ \sum_{i=1}^{I_\rho}\delta(w_i,\tilde{w}_L)\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda}) \\ \log\left(P(\mathcal{W})\right) \end{bmatrix} \qquad (3)$$

where $\{\tilde{w}_1,\dots,\tilde{w}_L\}$ are all the unique segment labels, $P(\mathcal{W})$ is given by language model, T is the number of frame in utterance $\mathcal{O}$, it is utilised to normalise the feature space corresponding the utterances with various length, and $\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda})$ is the log-likelihood feature vector, which can be described as follows:

$$\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda}) = \begin{bmatrix} \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_1})\right) \\ \vdots \\ \log\left(p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_L})\right) \end{bmatrix}_{\mathtt{L}\times 1} \qquad (4)$$

In equation (4), $p(\boldsymbol{O}_i|\boldsymbol{\lambda}_{\tilde{w}_l})$ is the likelihood of the HMM corresponding to label $\tilde{w}_l$ given the segment $\boldsymbol{O}_i$.

In terms of the large-margin training of the log-linear model, the margin is defined as the log-posterior ratio between the reference $\mathcal{W}_i$ and the most competing hypothesis $\mathcal{W}$. By introducing the prior $P(\boldsymbol{\eta})$ as a regularization term, the training criterion can be described as minimising:

$$\sum_{i=1}^{N}\left[\max_{\mathcal{W},\rho\neq\mathcal{W}_i,\rho_i}\left\{\mathcal{L}(\mathcal{W},\mathcal{W}_i) - \log\left(\frac{P(\mathcal{W}_i,\rho_i|\mathcal{O}_i,\boldsymbol{\eta},\boldsymbol{\lambda})}{P(\mathcal{W},\rho|\mathcal{O}_i,\boldsymbol{\eta},\boldsymbol{\lambda})}\right)\right\}\right]_+ - \log P(\boldsymbol{\eta}) \qquad (5)$$

When the prior $P(\boldsymbol{\eta})$ is given a Gaussian distribution $P(\boldsymbol{\eta}) = \mathcal{N}(\boldsymbol{\eta};\boldsymbol{\mu}_\eta,\boldsymbol{\Sigma}_\eta)$ with mean $\boldsymbol{\mu}_\eta$ and scaled identity covariance matrix $\boldsymbol{\Sigma}_\eta = C\boldsymbol{I}$, and substituting equation (2) into equation (5), then the large-margin training criterion can be further described as follows [10]:

$$\frac{1}{2}||\boldsymbol{\eta}-\boldsymbol{\mu}_{\boldsymbol{\eta}}||^2 + C\sum_{i=1}^{N}\left[\max_{\mathcal{W},\rho\neq\mathcal{W}_i,\rho_i}\left\{\boldsymbol{\eta}^{\mathsf{T}}\Phi(\mathcal{O}_i,\mathcal{W};\boldsymbol{\lambda},\rho) + \mathcal{L}(\mathcal{W},\mathcal{W}_i)\right\} - \boldsymbol{\eta}^{\mathsf{T}}\Phi(\mathcal{O}_i,\mathcal{W}_i;\boldsymbol{\lambda},\rho_i)\right]_+ \qquad (6)$$

4. INFINITE STRUCTURED SVM

The equation for the DP mixture of experts are given in equation (1). In order to apply this type of model to large vocabulary CSR, the structures need to be incorporated in the model. The direct way to introduce structure is incorporating the structures into the experts. If each expert is given a SSVM described in equation (2), then the DP mixture of experts given in equation (1) becomes:

$$P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\Theta,H) = \sum_{z\in\mathbb{Z}} P(z|\mathcal{O},\boldsymbol{\lambda},\Theta)P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\boldsymbol{\eta}_m) \qquad (7)$$

Here, the indicator variable z corresponds to the utterance $\mathcal{O}$, and the indicator set $\mathbb{Z}$ is infinitely sized. In the gating network, if the utterance is treated as a whole, the utterance indicator z is a scalar. The resulted model consists of infinite number of SSVMs, which is called infinite structured SVM (iSSVM); If the utterance is considered as being comprised of various segments, the utterance indicator z becomes a vector. The resulted model becomes a SSVM with infinite structures, which is called structured infinite SVM (SiSVM). In the following paper, the iSSVM will be detailedly discussed.

Suppose the training data are $\mathcal{D} = \{\mathcal{O}_1,\dots,\mathcal{O}_N;\mathcal{W}_1,\dots,\mathcal{W}_N;\rho_1,\dots,\rho_N\}$, as a Bayesian model, the classification of the iSSVM can be described as follows:

$$P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{D}) = \int P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{A})p(\mathcal{A}|\mathcal{D})d\mathcal{A} \qquad (8)$$

Since the integral in equation (8) is not intractable, the Monte Carlo Markov chain (MCMC) method is applied to approximate this integral, then the classification can be further described as follows:

$$\begin{aligned} P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{D}) &\approx \frac{1}{K}\sum_{k=1}^{K} P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{A}^{(k)}) \qquad (9)\\ &= \frac{1}{K}\sum_{k=1}^{K}\sum_{m=1}^{M_k} P(z=m|\mathcal{O},\boldsymbol{\lambda},\Theta^{(k)})P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\boldsymbol{\eta}_m^{(k)}) \end{aligned}$$

where $\mathcal{A} = \{\Theta, H\}$ are all the parameters of the iSSVM, and $\mathcal{A}^{(k)}$ are sampled from the model posterior distribution $p(\mathcal{A}|\mathcal{D})$. Here, K samples are used to approximate this intractable integral. Since $\mathcal{A}$ is the whole parameter set of the iSSVM, the joint posterior distribution $p(\mathcal{A}|\mathcal{D})$ do not have a closed form. Thus, Gibbs sampling [15] is used to obtain samples from this joint posterior distribution. In the sampling, the auxiliary variables $\boldsymbol{z} = \{z_1, \ldots, z_N\}$ (which are indicators of the training data) are introduced. The samples $\mathcal{A}^{(k)}$ are obtained by sampling from $p(\mathcal{A}, \boldsymbol{z}|\boldsymbol{\lambda}, \mathcal{D})$ yielding $\{\mathcal{A}^{(k)}, \boldsymbol{z}^{(k)}\}$ ignoring the samples $\boldsymbol{z}^{(k)}$, and $\mathcal{A}^{(k)}$ can be considered as being sampled from $p(\mathcal{A}|\boldsymbol{\lambda}, \mathcal{D})$ [16]. M_k is the number of unique values of the sampled indicators $\boldsymbol{z}^{(k)}$.

Θ is the parameter set of the gating network which is a DP mixture model here. The conditional posterior distribution of the parameter set can be described as follows:

$$p(\Theta|H^{(k)}, \boldsymbol{z}^{(k)}, \boldsymbol{\lambda}, \mathcal{D}) = p(\Theta|\{\phi(\mathcal{O}_n; \boldsymbol{\lambda})\}_{n=1}^{N}, \boldsymbol{z}^{(k)}) \qquad (10)$$

where $\phi(\mathcal{O}_n, \boldsymbol{\lambda})$ is the feature space for the utterance $\mathcal{O}_n$, which maps the observation $\mathcal{O}_n$ to a space with fixed dimension. The feature would be the log-likelihood feature of the whole utterance. Here, the normalised features based on segments are used: $\phi(\mathcal{O}_n, \boldsymbol{\lambda}) = 1/T_n \sum_i \varphi(\boldsymbol{O}_i, \boldsymbol{\lambda})$, where T_n is number of frames in utterance $\mathcal{O}_n$, and $\varphi(\boldsymbol{O}_i, \boldsymbol{\lambda})$ is the log-likelihood feature described in equation (4). Given the features $\{\phi(\mathcal{O}_n; \boldsymbol{\lambda})\}_{n=1}^{N}$and corresponding indicators $\boldsymbol{z}^{(k)}$, $\Theta^{(k)}$ can be sampled through the methods described in [12, 13].

In terms of the parameters of the experts H, each expert is given a log-linear model with large margin training, so the parameter of the m^{th} expert $\boldsymbol{\eta}_m$ is obtained through equation (6) with the data associated with expert m. Similar to the method used in [7], the mean of the prior $\boldsymbol{\mu}_{\boldsymbol{\eta}}$ is obtained from the SSVM trained on the whole training set. In the training of the iSSVM, if there are very few data associated with a expert, the trained expert might lack generalisation. Thus, each expert uses an informative prior with mean trained on the whole training set. By introducing this non-zero mean, the iSSVM should retrieve the performance of the SSVM, if C is small enough. Better performance could be achieved by gradually increasing C. 1-slack cutting plane algorithm [17] is utilised to train the SSVM, the constraint set for training the current SSVM parameter $\boldsymbol{\eta}_m^{(k)}$ can be cached and propagate to the next iteration of obtaining $\boldsymbol{\eta}_m^{(k+1)}$. This caching method can make the training more efficient, especially when applying the iSSVM to the large vocabulary CSR.

The indicator variable z_n is sampled according to the following posterior distribution:

$$\begin{aligned} &P(z_n = m|\mathcal{A}^{(k)}, \boldsymbol{z}_{-n}, \boldsymbol{\lambda}, \mathcal{D}) \propto \qquad (11)\\ &\quad P(z_n = m|\boldsymbol{z}_{-n}, \alpha)p\big(\phi(\mathcal{O}_n, \boldsymbol{\lambda})|\boldsymbol{\theta}_m^{(k)}\big)P(\mathcal{W}_n, \rho_n|\mathcal{O}_n, \boldsymbol{\lambda}, \boldsymbol{\eta}_m^{(k)}) \end{aligned}$$

where $\boldsymbol{z}_{-n}$ are all the indicators except z_n. The first term $P(z_n = m|\boldsymbol{z}_{-n}, \alpha)$ is given from the *Chinese Restaurant Process* (CRP) with concentration parameter α [18]. The term $P(\mathcal{W}_n, \rho_n|\mathcal{O}_n, \boldsymbol{\lambda}, \boldsymbol{\eta}_m^{(k)})$ is the posterior distribution given from the log-linear model described in equation (2), and term $p\big(\phi(\mathcal{O}_n, \boldsymbol{\lambda})|\boldsymbol{\theta}_m^{(k)}\big)$ is the component likelihood. When z_i indicates an existing expert, it is straightforward to calculate the conditional posterior distribution of z_n. When z_n denotes a new expert, following the method introduced in [12], in calculating the likelihood $p\big(\phi(\mathcal{O}_n, \boldsymbol{\lambda})|\boldsymbol{\theta}\big)$, the parameter $\boldsymbol{\theta}$ is sampled from its prior distribution as an auxiliary parameter, then the likelihood can be easily obtained. In order to make the newly generated expert having good generalisation, in calculating the third term, the parameter for the expert $\boldsymbol{\eta}$ is given as the the mean of its prior, namely the optimised parameter of the SSVM trained on the whole training set.

5. CLASSIFICATION

The equation used to classification has been given in equation (9), by substituting the log-linear model given in equation (2) into (9), the equation can be further described as follows:

$$\begin{aligned} &P(\mathcal{W}, \rho|\mathcal{O}, \boldsymbol{\lambda}, \mathcal{D}) \qquad (12)\\ &\approx \frac{1}{K}\sum_{k=1}^{K}\sum_{m=1}^{M_k} P(z_k = m|\mathcal{O}, \boldsymbol{\lambda}, \Theta^{(k)})\frac{\exp\big(\boldsymbol{\eta}_m^{(k)\mathsf{T}}\Phi(\mathcal{O}, \mathcal{W}; \boldsymbol{\lambda}, \rho)\big)}{\mathcal{S}_m^k} \end{aligned}$$

where $\mathcal{S}_m^k$ is the normalisation term:

$$\mathcal{S}_m^k = \sum_{\mathcal{W}', \rho'} \exp\big(\boldsymbol{\eta}_m^{(k)\mathsf{T}}\Phi(\mathcal{O}, \mathcal{W}'; \boldsymbol{\lambda}, \rho')\big) \qquad (13)$$

In the SSVM, this normalising constant can be ignored, since no posterior need to be calculated and the normalisation term stays the same for all the possible labels. In the iSVM, the posterior given by the log-linear model need to be calculated. Thus, this term cannot be ignored, but it is trivial to calculate, since the possible number of labels are small for each segment. In the iSSVM, the calculation of this term $\mathcal{S}_m^k$ is nontrivial, since the possible number of labels are exponentially large for the utterance $\mathcal{O}$.

Given that the possible number of labels are extremely large, the summation in equation (13) is quite inefficient. The forward algorithm can be adopted to calculate this summation efficiently on the lattice. According to the definition of the joint feature space given in equation (3), and supposing the parameter of the SSVM can be described as $\boldsymbol{\eta} = \{\boldsymbol{\eta}_{\tilde{w}_1}^{\mathsf{T}}, \ldots, \boldsymbol{\eta}_{\tilde{w}_L}^{\mathsf{T}}, \eta_{\mathcal{W}}\}^{\mathsf{T}}$, the normalisation term in equa-

System	Features	Test Set WER(%)			Avg
		testa	testb	testc	
HMM	—	9.83	9.11	9.53	9.48
SVM	Log-Like	8.29	7.90	8.61	8.20
iSVM		8.25	7.87	8.53	8.15
SSVM	Joint Feat	7.78	7.29	7.98	7.63
iSSVM		7.60	7.25	7.77	7.49

Table 1. The results on Aurora 2 database

tion (13) can be further described as follows:

$$\mathcal{S}_m^k = \sum_{\mathcal{W}',\rho'} \exp\Big(\frac{1}{T}\Big[\sum_{i=1}^{I_{\rho'}} \boldsymbol{\eta}_{m,w_i}^{(k)\mathsf{T}}\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda}) + \eta_{m,\mathcal{W}}\log P(\mathcal{W})\Big]\Big)$$
$$= \sum_{\mathcal{W}',\rho'}\Big[P(\mathcal{W})^{\frac{\eta_{m,\mathcal{W}}}{T}}\prod_{i=1}^{I_{\rho'}}\exp\big(\frac{1}{T}\boldsymbol{\eta}_{m,w_i}^{(k)\mathsf{T}}\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda})\big)\Big] \qquad (14)$$

Again, T is number of frames in utterisance $\mathcal{O}$, and $I_{\rho'}$ is the number of segments given the segmentation ρ', which is one path in the lattice. $P(\mathcal{W})$ is the probability of the word sequence. If the bigram language model is used here, the probability can be described as $P(\mathcal{W}) = \prod_{i=1}^{I_{\rho'}} P(w_i|w_{i-1})$, and here $P(w_1|w_0)$ is defined as $P(w_1|w_0) = P(w_1)$. Then, equation (14) can be further described as follows:

$$\mathcal{S}_m^k = \sum_{\mathcal{W}',\rho'}\Big\{\prod_{i=1}^{I_{\rho'}}\Big[P(w_i|w_{i-1})^{\frac{\eta_{m,\mathcal{W}}}{T}}\exp\big(\frac{1}{T}\boldsymbol{\eta}_{m,w_i}^{(k)\mathsf{T}}\varphi(\boldsymbol{O}_i;\boldsymbol{\lambda})\big)\Big]\Big\} \qquad (15)$$

Then the forward algorithm can be applied to calculate this summation on the lattice. At each node in the lattice, the scores are merged. The detail of the forward algorithm is discussed in [19].

By introducing the forward algorithm on the lattice, $\mathcal{S}_m^k$ can be calculated efficiently. The computational complexity becomes $\mathrm{O}(N_{\text{arc}}L)$, where N_{arc} is number of arcs in the lattice, and L is the unique number of segment labels. Normally, the arcs in a lattice are reasonably sized. For Aurora 2 data set, there are around several hundred arcs in a lattice.

In the classification, the best alignment ρ and corresponding word sequence $\mathcal{W}$ need to be found through equation (12). But, how to find the path ρ and corresponding $\mathcal{W}$, which maximise the posterior $P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{D})$, might be a problem here. In the structured SVM, the Viterbi algorithm is applied to solve this problem [10]. However, in the iSSVM, the parameter $\boldsymbol{\eta}_m^{(k)}$ varies with experts m and samples k. Only when the m and k are given, the Viterbi algorithm can be applied, but the best alignment ρ and corresponding word sequence $\mathcal{W}$ might change with different $\boldsymbol{\eta}_m^{(k)}$. Thus, the Viterbi algorithm cannot be directly applied, and an approximation is made here. Rather than calculating the posterior $P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{D})$ by enumerating all the possible ρ and $\mathcal{W}$ which could be exponentially large, the Viterbi algorithm is implemented to keep the N-best alignments [1] ρ and corresponding labels $\mathcal{W}$ in set $\mathbb{P}$ for all k and m. After the set $\mathbb{P}$ is obtained, the classification can be described as follows:

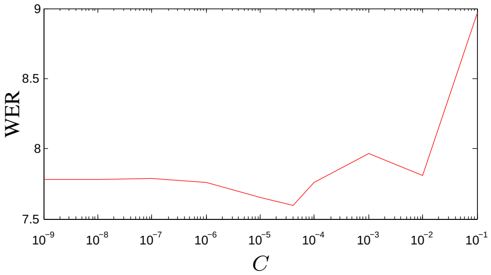

Fig. 2. The iSSVM performance on set A with different C

$$\{\mathcal{W},\rho\} \approx \arg\max_{\mathcal{W},\rho} P(\mathcal{W},\rho|\mathcal{O},\boldsymbol{\lambda},\mathcal{D}) \qquad \forall\{\mathcal{W},\rho\}\in\mathbb{P} \qquad (16)$$

6. EXPERIMENTS AND CONCLUSIONS

The performance of the proposed iSSVM is evaluated on the Aurora 2 database [20]. The utterances in this database are continuous digit strings with vocabulary size 12 (one to nine, plus zero, oh and silence). The generative models (HMMs) are trained on the clean data with 8840 utterances. The noise model for VTS compensation [21] is estimated on each utterance, and the performance of the VTS compensated HMM is listed in Table 1. The SVM, iSVM , SSVM a nd iS SVM 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 the experiments, the log-likelihood features described in equation (4) are used by the SVM and iSVM, and the joint features described in equation (3) are used by the SSVM and iSSVM. The results are listed in Table 1. On test set A and B, the iSSVM get around 3% relative improvement in all SNRs, but quite small improvement is achieved on test set B. The large margin training criterion described in equation (6) is adopted to train the experts (SSVM) of the iSSVM, and different experts share the same C. The parameter C is tuned on the test set A illustrated in Fig 2. Since the prior mean $\boldsymbol{\mu}_{\boldsymbol{\eta}}$ in equation (6) is given the optimised parameter of the SSVM trained on the whole training set, when C is small, the SSVM performance is retrieved, and the optimised C can be found by increasing C.

This paper has studied the iSSVM which is a direct extension of the iSVM described in previous paper [7]. Taking advantage of the mixture of experts and structured model, the iSSVM outperforms the iSVM and SSVM. As being discussed in Section 4, the utterance indicator of the iSSVM is a scalar, which means all the segments in an utterance share the same indicator. This might limit the flexibility of the gating network. Moreover, the distribution of the utterances are explored by the gating network, and it is hard to model the utterance distribution compared with segments. This may limit system performance. Thus, future work will study the model called structured infinite SVM (SiSVM) which is a structured model and deploys a more flexible and reasonable gating network.

[1] In this paper, only the 1-best alignment is considered.

7. REFERENCES

[1] 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.

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

[3] 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.

[4] Yongqiang Wang and Mark Gales, "Speaker and noise factorization for robust speech recognition," *IEEE Transactions on Audio, Speech and Language Processing*, vol. 20, no. 7, pp. 2149–2158, 2012.

[5] 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.

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

[7] Jingzhou Yang, Rogier C van Dalen, and Mark Gales, "Infinite support vector machines in speech recognition," in *Proceedings of Interspeech*, Lyon, France, 2013, pp. 3303–3307.

[8] Veera Venkataramani, Shantanu Chakrabartty, and William Byrne, "Support vector machines for segmental minimum bayes risk decoding of continuous speech," in *IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*, 2003, pp. 13–18.

[9] 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)*, New York, NY, USA, 2004, pp. 104–111.

[10] 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.

[11] Shi-Xiong Zhang and Mark Gales, "Kernelized log linear models for continuous speech recognition," in *ICASSP*, 2013, pp. 6950–6954.

[12] 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.

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

[14] Shi-Xiong Zhang and Mark Gales, "Structured support vector machines for noise robust continuous speech recognition," in *Proceedings of Interspeech*, Florence, Italy, 2011, pp. 989–992.

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

[16] Christophe Andrieu, Nando De Freitas, Arnaud Doucet, and Michael I. Jordan, "An introduction to MCMC for machine learning," *Machine Learning*, vol. 50, pp. 5–43, 2003.

[17] Thorsten Joachims, Thomas Finley, and Chun-Nam John Yu, "Cutting-plane training of structural SVMs," *Machine Learning*, vol. 77, no. 1, pp. 27–59, 2009.

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

[19] Xuedong Huang, Alejandro Acero, Hsiao-Wuen Hon, et al., *Spoken language processing*, vol. 15, Prentice Hall PTR New Jersey, 2001.

[20] 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.

[21] 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, China, 2000, pp. 869–872.